

SINTONIA DE CONTROLADORES BASEADA EM SISTEMAS DE INFERÊNCIA DIFUSA

Nuno Filipe Lopes Agualusa Pires



Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

2012

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha da Unidade Curricular de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de Computadores

Candidato: Nuno Filipe Lopes Agualusa Pires, N° 1040115, 1040115@isep.ipp.pt

Orientação científica: Isabel Maria de Sousa de Jesus, ISJ@isep.ipp.pt



Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

7 de Novembro de 2012

Agradecimentos

Durante esta etapa do meu percurso acadêmico, foram muitos os momentos que sem o apoio de algumas pessoas não seria possível chegar até aqui.

Agradeço à Professora Isabel Jesus pelo apoio constante que me transmitiu, pela disponibilidade com que me ajudou a superar os obstáculos e pela orientação dada no sentido de chegar a bom porto.

Agradeço à minha família e namorada, pois foram incansáveis desde o princípio. A sua persistência foi vital para o meu sucesso.

Quero também deixar uma palavra de agradecimento ao meu amigo e colega Jorge Barbosa pelo seu companheirismo e amizade ao longo desta etapa.

Resumo

Este trabalho de pesquisa e desenvolvimento tem como fundamento principal o Conceito de Controlo por Lógica Difusa. Utilizando as ferramentas do *software* Matlab, foi possível desenvolver um controlador com base na inferência difusa que permitisse controlar qualquer tipo de sistema físico real, independentemente das suas características.

O Controlo Lógico Difuso, do inglês “Fuzzy Control”, é um tipo de controlo muito particular, pois permite o uso simultâneo de dados numéricos com variáveis linguísticas que tem por base o conhecimento heurístico dos sistemas a controlar. Desta forma, consegue-se quantificar, por exemplo, se um copo está “meio cheio” ou “meio vazio”, se uma pessoa é “alta” ou “baixa”, se está “frio” ou “muito frio”.

O controlo PID é, sem dúvida alguma, o controlador mais amplamente utilizado no controlo de sistemas. Devido à sua simplicidade de construção, aos reduzidos custos de aplicação e manutenção e aos resultados que se obtêm, este controlador torna-se a primeira opção quando se pretende implementar uma malha de controlo num determinado sistema. Caracterizado por três parâmetros de ajuste, a saber componente proporcional, integral e derivativa, as três em conjunto permitem uma sintonia eficaz de qualquer tipo de sistema.

De forma a automatizar o processo de sintonia de controladores e, aproveitando o que melhor oferece o Controlo Difuso e o Controlo PID, agrupou-se os dois controladores, onde em conjunto, como poderemos constatar mais adiante, foram obtidos resultados que vão de encontro com os objectivos traçados.

Com o auxílio do *simulink* do Matlab, foi desenvolvido o diagrama de blocos do sistema de controlo, onde o controlador difuso tem a tarefa de supervisionar a resposta do controlador PID, corrigindo-a ao longo do tempo de simulação. O controlador desenvolvido é denominado por Controlador *FuzzyPID*.

Durante o desenvolvimento prático do trabalho, foi simulada a resposta de diversos sistemas à entrada em degrau unitário. Os sistemas estudados são na sua maioria sistemas físicos reais, que representam sistemas mecânicos, térmicos, pneumáticos, eléctricos, etc.,

e que podem ser facilmente descritos por funções de transferência de primeira, segunda e de ordem superior, com e sem atraso.

Palavras-Chave

Lógica Difusa, PID, Matlab, *Simulink*, Função de Transferência, Sistemas Físicos.

Abstract

This development and research thesis have as principal concept the Fuzzy Logic Control. Using software tools of Matlab was possible to develop a controller based on Fuzzy Inference which allowed controlling any kind of real physical system, regardless their characteristics.

The Fuzzy Logic Controller is a very particular type of controller, as it allows the simultaneous use of numeric data with linguist variables that is based on heuristic knowledge on the controlling systems. Thus, it enables to quantify, for example, if a glass is “half full” or “half empty”, if a person is “tall” or “short” or if it is “cold” or “very cold”.

The PID Controller is, without doubt, the most widely used controller in the control systems. Due to its simplicity of construction, the low cost implementation, maintenance and the good results that are obtained, this controller makes the first option when you intend to implement a control loop in a given system. Characterized by three adjusting parameters, which are proportional, integral and derivative components, those together allow efficient tuning of any type of system.

In order to automate the process of tuning control and using what Fuzzy and PID Controllers best offer, grouped the two controllers, where together, as will be presented further, the obtained results meet the defined objectives.

With the suport of Matlab Simulink, was developed a block diagram of the control system, where the Fuzzy Controller has the task of supervising the response of the PID Controller, correcting it during the simulation. The developed controller was named by FuzzyPID Controller.

During the pratical development was simulated the response of several systems to unit step input. The systems studied are mostly real physical systems, which represent mechanical, thermal, pneumatic, electrical, etc, and they can be easily described by a transfer functions of first, second and higher order, with and without time delay.

Keywords

Fuzzy Logic, PID, Matlab, Simulink, Transfer Function, Physical Systems.

Índice

AGRADECIMENTOS	I
RESUMO	III
ABSTRACT	V
ÍNDICE	VII
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABELAS	XII
ACRÓNIMOS.....	XV
1. INTRODUÇÃO	1
1.1. CONTEXTUALIZAÇÃO	2
1.2. OBJECTIVOS.....	3
1.3. CALENDARIZAÇÃO	4
1.4. ORGANIZAÇÃO DO RELATÓRIO	4
2. CONTROLO LÓGICO DIFUSO.....	7
2.1. NOTA HISTÓRICA.....	8
2.2. APLICAÇÕES PRÁTICAS.....	9
2.3. CONJUNTOS DIFUSOS E LÓGICA DIFUSA: A BASE DO CONTROLO DIFUSO.....	10
2.4. LÓGICA DIFUSA	16
2.5. ARQUITECTURA DE UM SISTEMA DIFUSO	22
2.6. CONCLUSÕES	30
3. ANÁLISE DA RESPOSTA DE SISTEMAS	31
3.1. INTRODUÇÃO	31
3.2. SISTEMAS DE 1º ORDEM.....	39
3.3. SISTEMAS DE 2º ORDEM.....	41
3.4. SISTEMAS DE ORDEM SUPERIOR	48
3.5. O EFEITO DOS ZEROS NA RESPOSTA DO SISTEMA	50
3.6. CONCLUSÕES	51
4. CONTROLADOR PID	53
4.1. NOTA HISTÓRICA SOBRE O PID	53
4.2. INTRODUÇÃO AO CONTROLO PID	54
4.3. ESCOLHA DO TIPO DE CONTROLADOR.....	60
4.4. MÉTODO PRÁTICOS DE SINTONIA DE CONTROLADORES PID	61
4.5. CONCLUSÕES	63

5. DESENVOLVIMENTO DO CONTROLADOR DIFUSO.....	65
5.1. SISTEMA DE CONTROLO	66
6. RESULTADOS OBTIDOS.....	79
6.1. SISTEMAS SEM ATRASO	80
6.2. SISTEMAS COM ATRASO	90
6.3. CONCLUSÕES.....	106
7. CONCLUSÕES	107
REFERÊNCIAS DOCUMENTAIS.....	109

Índice de Figuras

Figura 1 - Representação de operações de conjuntos utilizando Diagramas de Venn [MIRANDA]	11
Figura 2 - Funções de Pertença para valores pequenos, médios e grandes [MIRANDA]	13
Figura 3 - Operação de União entre dois Conjuntos Difusos A e B [MIRANDA]	14
Figura 4 - Operação de Intersecção entre dois Conjuntos Difusos A e B [MIRANDA]	15
Figura 5 - Operação de Negação de dois Conjuntos Difusos A e B [MIRANDA]	16
Figura 6 - Conjunto de funções de pertença da variável Temperatura [MIRANDA]	17
Figura 7 – Valores linguísticos da variável Velocidade [MIRANDA]	18
Figura 8 - Funções de pertença do pêndulo invertido [MIRANDA]	19
Figura 9 - Matriz de regras para o pêndulo invertido [MIRANDA]	21
Figura 10 - Arquitectura de um Controlador Lógico Difuso [KEVIN]	22
Figura 11 - Conjuntos Difusos da variável Temperatura [MIRANDA]	23
Figura 12 - Tipos de Funções de Pertença. (a) trapezoidal, (b) gaussiana, (c) pico, (d) triangular [KEVIN]	24
Figura 13 - Método de Cálculo dos Graus de Pertença [MIRANDA]	25
Figura 14 - Sistema Difuso de mercado de acções [MIRANDA]	26
Figura 15 - Colapsagem - Método de Centro de Área [MIRANDA]	28
Figura 16 - Colapsagem - Método da Média dos Máximos MOM [MIRANDA]	29
Figura 17 – Diagrama típico de um sistema de processamento difuso [MIRANDA]	29
Figura 18 - Sistema de Controlo de uma Antena [TROFINO]	33
Figura 19 - Curvas típicas da resposta ao degrau unitário [TROFINO]	34
Figura 20 - Parâmetros de desempenho resultantes da entrada em degrau [TROFINO]	35
Figura 21 - Localização de pólos no plano-s [ANALYSIS]	38
Figura 22 - Sistema de Primeira Ordem [HELIO]	39
Figura 23 - Localização de um Pólo no plano-s de um Sistema de 1º Ordem [SANTOS]	40
Figura 24 - Sistema de Primeira Ordem [SANTOS]	40
Figura 25 - Curva de Resposta à entrada em Degrau Unitário [SANTOS]	41
Figura 26 - Sistema de Segunda Ordem [HELIO]	41
Figura 27 - Localização do Pólos num Sistema Subamortecido [EI]	43
Figura 28 – Resposta de um Sistema Sub-amortecido à entrada em Degrau [EI]	43
Figura 29 - Localização dos Pólos de um Sistema não amortecido [EI]	44
Figura 30 - Resposta de um sistema de segunda ordem à entrada em degrau não amortecido [EI]	45
Figura 31 - Localização dos Pólos de um Sistema onde o coeficiente de amortecimento é unitário [EI]	45

Figura 32 - Resposta de um sistema de segunda ordem à entrada em degrau criticamente amortecido [EI].....	46
Figura 33 - Localização dos Pólos de um Sistema sobre-amortecido [EI].....	46
Figura 34 - Resposta de um sistema de segunda ordem à entrada em degrau com sobre-amortecimento [EI].....	47
Figura 35 – Variação da resposta de um Sistema de Segunda Ordem com a variação do factor de amortecimento [EI].....	48
Figura 36 - Sistema de Ordem Superior [HELIO]	48
Figura 37 - Resposta de diversos sistemas à entrada em degrau variando os zeros da função transferência [MARUYAMA]	51
Figura 38 - Diagrama de blocos do Controlador PID [http://en.wikipedia.org/wiki/File:PID_en.svg]	55
Figura 39 - Diagrama de Blocos de um Controlador P [LOURENÇO]	56
Figura 40 - Diagrama de Blocos de um Controlador PI [LOURENÇO].....	56
Figura 41 - Curva da resposta de saída para um controlador PI [LOURENÇO]	57
Figura 42 -Diagrama de blocos de um controlador PD [LOURENÇO].....	58
Figura 43 - Curva da resposta de saída para um controlador PD [LOURENÇO].....	58
Figura 44 - Diagrama de blocos de um controlador PID [LOURENÇO]	59
Figura 45 - Resposta ao degrau unitário para o primeiro método de Ziegler-Nichols [YASSER] ..	61
Figura 46 - Resposta oscilatória obtida através do segundo método de ZN [YASSER].....	63
Figura 47 - Modelo do Controlador <i>Fuzzy</i> -PID [BOILER]	66
Figura 48 - Diagrama de Blocos que ilustra o Sistema de Controlo utilizado	67
Figura 49 - Bloco " <i>Step</i> "	68
Figura 50 – Funções matemáticas representadas através de blocos	68
Figura 51 - Visualização da resposta de um sistema através do <i>Scope</i>	69
Figura 52 - Entradas e Saídas do Controlador Difuso	70
Figura 53 - Definição da variável de entrada Erro, e	71
Figura 54 - Definição da variável de entrada Derivada do Erro, \dot{e}	72
Figura 55 - Definição das variáveis de saída K_P , K_I , K_D	73
Figura 56 - Controlador PID	78
Figura 57 - Análise temporal de um sistema de controlo de nível num tanque	80
Figura 58 - Análise temporal de um sistema hidráulico.....	81
Figura 59 - Análise temporal de um sistema mecânico de uma mola	82
Figura 60 - Análise temporal do sistema eléctrico RLC	83
Figura 61 - Análise temporal do sistema de controlo da posição de um motor DC	84
Figura 62 - Análise temporal do sistema de controlo da velocidade de um motor DC	85
Figura 63 - Análise temporal do primeiro sistema de 3ª ordem	86
Figura 64 - Análise temporal do segundo sistema de 3ª ordem.....	88
Figura 65 - Análise temporal do terceiro sistema de 3ª ordem.....	89

Figura 66 - Análise temporal do primeiro sistema de 1ª ordem com atraso de 5s	90
Figura 67 - Análise temporal do primeiro sistema de 1ª ordem com atraso de 30s	91
Figura 68 - Análise temporal do segundo sistema de 1ª ordem com atraso de 5s.....	92
Figura 69 - Análise temporal do segundo sistema de 1ª ordem com atraso de 30s.....	93
Figura 70 - Análise temporal do terceiro sistema de 1ª ordem com atraso de 30s.....	95
Figura 71 - Análise temporal do sistema mecânico de uma mola com atraso de 10s	96
Figura 72 - Análise temporal do sistema mecânico de uma mola com atraso de 30s	97
Figura 73 - Análise temporal do sistema eléctrico RLC série com atraso de 30s	98
Figura 74 - Análise temporal do sistema de controlo da posição de um motor DC com atraso de 30s	99
Figura 75 - Análise temporal do sistema de controlo da velocidade de um motor DC com atraso de 10s	100
Figura 76 - Análise temporal do sistema de controlo da velocidade de um motor DC com atraso de 10s	101
Figura 77 - Análise temporal do primeiro sistema de 3ª ordem com atraso de 50s	102
Figura 78 - Análise temporal do segundo sistema de 3ª ordem com atraso de 50s.....	103
Figura 79 - Análise temporal do terceiro sistema de 3ª ordem com atraso de 50s.....	104
Figura 80 - Análise temporal do quarto sistema de 3ª ordem com atraso de 50s.....	105

Índice de Tabelas

Tabela 1 - Calendarização do Projecto	4
Tabela 2 - Breve resumo histórico da tecnologia difusa [REZNIK]	8
Tabela 3 - Tabela de funções de transferência com zeros diferentes [MARUYAMA].....	50
Tabela 4 - Resumo dos efeitos de cada parâmetro P, I e D [LOURENÇO].....	59
Tabela 5 - Regras de Sintonia de Ziegler-Nichols (Primeiro Método) [YASSER].....	62
Tabela 6 - Regras de Sintonia de Ziegler-Nichols (Segundo Método) [LOURENÇO]	63
Tabela 7 - Regras difusas para a variável de saída KP1 (sem atraso)	74
Tabela 8 - Regras difusas para a variável de saída KI1 (sem atraso)	74
Tabela 9 - Regras difusas para a variável de saída KD1 (sem atraso).....	75
Tabela 10 - Regras difusas para a variável de saída KP1 (com atraso).....	76
Tabela 11 - Regras difusas para a variável de saída KI1 (com atraso).....	76
Tabela 12 - regras difusas para a variável de saída KD1 (com atraso)	77
Tabela 13 - Valores dos parâmetros temporais do sistema nível num tanque.....	81
Tabela 14 - Valores dos parâmetros temporais do sistema hidráulico	82
Tabela 15 - Valores dos parâmetros temporais do sistema mecânico de uma mola.....	83
Tabela 16 - Valores dos parâmetros temporais do sistema eléctrico RLC Série.....	84
Tabela 17 - Valores dos parâmetros temporais do sistema de controlo da posição de um motor DC	85
Tabela 18 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC	86
Tabela 19 - Valores dos parâmetros temporais do primeiro sistema de 3ª ordem.....	87
Tabela 20 - Valores dos parâmetros temporais do segundo sistema de 3ª ordem	88
Tabela 21 - Valores dos parâmetros do terceiro sistema de 3ª ordem.....	89
Tabela 22 - Valores dos parâmetros temporais para o primeiro sistema de 1ª ordem com atraso de 5s	91
Tabela 23 - Valores dos parâmetros temporais do primeiro sistema de 1ª ordem com atraso de 30s	92
Tabela 24 - Valores dos parâmetros temporais do segundo sistema de 1ª ordem com atraso de 5s	93
Tabela 25 - Valores dos parâmetros temporais do segundo sistema de 1ª ordem com atraso de 30s	94
Tabela 26 - Valores dos parâmetros temporais do terceiro sistema de 1ª ordem com atraso de 30s	95
Tabela 27 - Valores dos parâmetros temporais do sistema mecânico de uma mola com atraso de 10s	96

Tabela 28 - Valores dos parâmetros temporais do sistema mecânico de uma mola com atraso de 30s	97
Tabela 29 - Valores dos parâmetros temporais do sistema eléctrico RLC série com atraso de 30s.	98
Tabela 30 - valores dos parâmetros temporais do sistema de controlo da posição de um motor DC com atraso de 30s	99
Tabela 31 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC com atraso de 10s.....	100
Tabela 32 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC com atraso de 10s.....	102
Tabela 33 - Valores dos parâmetros temporais do primeiro sistema de 3ª ordem com atraso de 50s	103
Tabela 34 - Valores dos parâmetros temporais do segundo sistema de 3ª ordem com atraso de 50s	104
Tabela 35 - Valores dos parâmetros temporais do terceiro sistema de 3ª ordem com atraso de 50s	105
Tabela 36 . Valores dos parâmetros temporais do quarto sistema de 3ª ordem com atraso de 50s	106

Acrónimos

GUIDE – *Graphical User Interface Design Environment*

PID – Proporcional, Integral e Derivativo

COA – *Center of Area*

MOM – *Mean of maximum*

SISO – *Single Input and single Output*

1. INTRODUÇÃO

À escala global o controlo de sistemas tem proporcionado um significativo avanço da engenharia e da ciência. Para além da sua aplicabilidade, extremamente crucial, na ciência espacial, no controlo direccionado de mísseis e em sistemas robóticos, entre outros, o controlo automático tem sido fundamental para o controlo industrial e de produção. Com os avanços diários nesta área da engenharia, tem sido possível otimizar o desempenho de processos, melhorar a produtividade das fábricas e dos trabalhadores, diminuir o tempo de execução de diversas tarefas, bem como, diminuir o nível de dificuldade das mesmas e, ainda mais importante com a introdução de sistemas de controlo automático, consegue-se uma diminuição drástica da probabilidade de ocorrência de acidentes por falha humana.

Relativamente ao controlo de sistemas físicos, sendo eles mecânicos, térmicos, eléctricos, pneumáticos, hídricos e hidráulicos, entre outros, estes proporcionam grandes desafios no que respeita ao seu controlo e estabilização, onde pequenas alterações no processo onde estão inseridos, ou pequenas alterações nos seus parâmetros, originam grandes variações nas suas respostas de saída. Todos os sistemas físicos podem ser descritos matematicamente por funções de transferência de primeira, segunda e de ordem superior, com ou sem atraso, que são obtidas através das leis da física que regem determinado sistema, por exemplo, as leis de Newton para sistemas mecânicos e as leis de Kirchhof para sistemas eléctricos.

1.1. CONTEXTUALIZAÇÃO

Este trabalho surge no âmbito da unidade curricular de Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Eletrotécnica e de Computadores, tendo como tema proposto o Desenvolvimento de um Controlador com base em Sistemas de Inferência Difusa.

Numa fase inicial foi efectuado um estudo exaustivo sobre o conceito lógica difusa, mais especificamente o que caracteriza um controlador baseado em lógica difusa, as suas vantagens e desvantagens, como se desenvolve um controlador baseado na lógica difusa, a definição das suas variáveis de entrada e de saída e, como se transfere o conhecimento heurístico dos sistemas a controlar para o interior do controlador.

Posteriormente, realizou-se uma abordagem teórica ao Controlador PID, onde se estudou o efeito que cada parâmetro tem na resposta do controlador, tendo-se verificado que existem diversas formas de os sintonizar, verificou-se também as suas vantagens e desvantagens e as suas principais aplicações.

O controlador desenvolvido neste trabalho é a junção de um controlador lógico difuso com um controlador PID, por isso foi necessário definir a sua configuração para que o controlador lógico funcionasse como “supervisor” do controlador PID, ajustando automaticamente os seus parâmetros P,I e D consoante a resposta na saída do sistema.

Com a estrutura do controlador bem definida, seguiu-se uma fase posterior que seria decidir qual ou quais os sistemas que o controlador iria controlar, pelo que, optou-se por controlar sistemas reais. Sistemas mecânicos, eléctricos, térmicos, pneumáticos, entre outros, definidos matematicamente por funções de transferência, onde se verificou que as diferenças entre as funções de transferência os sistemas estariam apenas nos parâmetros que os definem.

Para conseguir pôr em prática o desenvolvimento do controlador foi necessário recorrer ao *Simulink* do Matlab, que permite a simulação de um processo de controlo através de diagrama de blocos. Estes mesmos diagramas de blocos têm cada um a sua função, que quando organizados e parametrizados numa ordem específica representam o processo a simular.

1.2. OBJECTIVOS

De uma forma geral o objectivo deste trabalho é aprofundar os conhecimentos em torno do controlo de sistemas, nomeadamente no que concerne aos métodos de controlo de sistemas através de Lógica Difusa e PID e, aprofundar a componente teórica sobre Teoria de Sistemas, o que se torna fundamental para compreender o comportamento dos diversos sistemas quando estimulados. O estudo sobre a Teoria de Sistemas, é a chave do sucesso do controlador, porque apenas se consegue bons resultados quando existe um conhecimento do sistema que se está a controlar.

- Dada a complexidade inerente a este objectivo, sentiu-se a necessidade de o subdividir em múltiplas tarefas de realização mais simples, tais como:
- Estudo teórico sobre teoria de sistemas;
- Estudo teórico e prático sobre o controlador lógico difuso;
- Estudo teórico e prático sobre o controlador PID;
- Estudo do software Matlab, nomeadamente o Simulink;
- Desenvolvimento de toda a estrutura do controlador;
- No final, após alcançar os objectivos traçados, o controlador será capaz de controlar qualquer sistema físico que o utilizador defina.

1.3. CALENDARIZAÇÃO

Na tabela seguinte é apresentada a calendarização, ilustrando como foi desenvolvido o trabalho ao longo do tempo.

Tabela 1 - Calendarização do Projecto

Calendarização do Projecto																
Tarefas	Ago	Set	Out	Nov	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov
Estruturação do trabalho a desenvolver																
Estudo teórico sobre sistemas																
Estudo da Lógica Difusa (Estado da Arte)																
Estudo do Controlador PID																
Estudo do <i>software</i> Matlab, Simulink, GUIDE																
Desenvolvimento prático do controlador Fuzzy-PID																
Obtenção de resultados e análise																
Elaboração do Relatório final																
Apresentação final																

1.4. ORGANIZAÇÃO DO RELATÓRIO

O presente relatório está organizado em 7 capítulos.

No Capítulo 1 é feita a introdução ao trabalho, a sua contextualização e calendarização, e por último traçados os objectivos.

No Capítulo 2 é apresentada a teoria que fundamenta a Lógica Difusa. Inicia-se o capítulo com uma nota histórica sobre o controlo através de Lógica Difusa e são enumerados alguns exemplos de aplicações mundanas. Os restantes subcapítulos estão focados em explicitar toda a teoria que envolve a Lógica Difusa.

No Capítulo 3 é abordado toda a teoria de sistemas que descreve o comportamento dos sistemas de primeira, segunda e ordem superior, com e sem atraso. Este capítulo serve principalmente como base para se compreender alguns fenómenos que ocorrem durante as diversas simulações realizadas.

O Capítulo 4 é dedicado ao controlador PID. Aqui são descritos os conceitos teóricos e práticos que o caracterizam. São enumerados alguns métodos de sintonia e em que

situações os aplicar e são apresentados os efeitos que cada parâmetro tem sobre os sistemas a controlar.

No Capítulo 5 é feita a exposição de toda a componente prática do trabalho. Começando pelo Matlab, é descrita a forma como foi definido o diagrama de blocos do controlador, através do *Simulink*. De seguida, a constituição e parametrização do controlador lógico difuso através das variáveis de entrada e de saída e as funções de pertença, seguindo-se a definição das regras que guiam o controlador.

No Capítulo 6 são apresentados todos os resultados obtidos na forma gráfica juntamente com os devidos comentários.

O Capítulo 7 é referente às conclusões. Aqui são feitos comentários gerais aos resultados obtidos, verificando-se se estes foram de encontro aos objectivos inicialmente traçados. São também enumerados alguns pontos a melhorar em desenvolvimentos futuros para este tipo de aplicação.

2. CONTROLO LÓGICO DIFUSO

De uma forma simplificada, podemos definir o conceito de lógica difusa como uma lógica multivalores, que permite a definição de grandezas através de valores intermédios que estejam entre os valores convencionais como verdadeiro/falso, sim/não, alto/baixo, etc. Esta propriedade permite que palavras ou frases habitualmente pronunciadas por humanos possam ser matematicamente definidas, processadas e associadas a valores de controlo [HELLMANN].

Por exemplo, num dia agradável de Verão uma afirmação como “está muito calor” pode ser simultaneamente verdadeira e falsa, dependendo de quem a avalia. Trata-se de uma afirmação qualitativa que representa uma opinião ao invés de um facto. Contudo, se for atribuído um valor quantitativo à afirmação “está muito calor”, esta passará a fazer sentido aos “olhos” de um controlador. Será vantajoso utilizar esta lógica em controlo? A resposta tanto poderá ter uma visão comercial como prática. Comercialmente, a lógica difusa tem sido usada com grande sucesso em controlo de máquinas industriais, em electrónica de consumo, equipamentos domésticos, etc. Na maioria das aplicações os sistemas lógicos difusos são simples de projectar, podem ser compreendidos e implementados facilmente

por leigos em teoria de controlo. Claro que o sistema de controlo poderá não ser óptimo mas será aceitável [JAMES VERNON].

2.1. NOTA HISTÓRICA

O conceito de lógica difusa foi criado em 1965 por Lotfi A. Zadeh, professor de ciências de computação na universidade da Califórnia em Berkeley. Depois disso, tem surgido grandes avanços nesta área de controlo, os quais são descritos na Tabela 2.

Tabela 2 - Breve resumo histórico da tecnologia difusa [REZNIK]

1965	Concepção do conceito de lógica difusa por Lotfi Zadeh (USA)
1972	Criado o primeiro grupo de trabalho em sistemas difusos, no Japão por Toshiro Terano
1973	Lançado um artigo sobre algoritmos difusos por Zadeh (EUA)
1974	Controlo de um motor a vapor por Ebrahim Mamdani (UK)
1977	Primeiro sistema difuso criado para avaliação de empréstimos por Hans Zimmermann (Alemanha)
1980	Controlo de estufa de secagem de cimento por F. – L- Smidth & Co. – Lauritz P. Holmblad (Dinamarca) Jogos de Xadrez e de Gamão baseados na lógica difusa, criado por Hans Berliner (EUA)
1984	Controlo de um sistema de tratamento de água (injecção química) Sistema de controlo do metro <i>Sendai Transportation</i> (Japão)
1985	Primeiro chip difuso desenvolvido por Masaki Togai e Hiroyuke Watanabe na Bell Labs (EUA)
1986	Sistema difuso para diagnóstico de doenças na Omron (Japão)
1987	Robô de soldadura Sistema de aterragem de uma aeronave

	Segunda conferência IFSA em Tokyo Togai InfraLogic Inc. - Primeira empresa a trabalhar com a Lógica Difusa, Irvine (EUA)
1988	Controlo de uma estufa pela Yokogawa Primeiro controlador difuso dedicado vendido – Omron (Japão)
1989	Criação do Laboratório Internacional para pesquisa na área da Engenharia Difusa no Japão
1990	Televisão difusa pela Sony (Japão) “olho” electrónico difuso pela Fujitsu (Japão) Instituto da Lógica Difusa por Takeshi Yamakawa (Japão) Laboratório de Controlo de Sistemas Inteligentes na Siemens (Alemanha)
1991	Centro de promoção da Lógica Difusa (Japão) Kit educacional pela Motorola (EUA)
1992 até aos dias de hoje	Continuação da evolução da Lógica Difusa com o aparecimento de inúmeros projectos de sucesso.

2.2. APLICAÇÕES PRÁTICAS

A partir de 1991 a tecnologia difusa saiu dos laboratórios para ser aplicada na indústria, tornando-se uma ferramenta muito útil. Um pequeno resumo de alguns projectos de sucesso é apresentado de seguida. Será perceptível a enorme variedade de aplicações em que se utiliza a Lógica Difusa. [REZNIK]

- Controlo automático das comportas de uma barragem hidroeléctrica (Tokyo Electric Power);
- Controlo de movimentos mecânicos de róbos (Hirote, fuji Electric, Toshiba, Omron);
- Controlo automático da direcionalidade das câmaras de vídeo em eventos desportivos;
- Controlo de eficiência e estabilização de motores de automóveis (Nissan);
- Controlo de velocidade de automóveis (Nissan, Subaru);
- Substituição de um perito (corrector de bolsa) de apoio a actividades económicas (Yamaichi, Hitachi);

- Optimização dos horários de transportes públicos (Toshiba, NipponSystem, Keihan-Express);
- Sistema automatizado de arquivo de documentos (Mitsubishi Electric);
- Sistema de previsão de tremores de terra (Seismology Bureau of Metrology, Japan);
- Sistema de apoio ao diagnóstico do cancro (Kawasaki Medical School);
- Reconhecimento de padrões em imagens obtidas com cameras de vídeo (Canon, Minolta);
- Controlo automático de potência de aspiradores, baseados no reconhecimento do tipo e estado da superfície (Matsushita);
- Controlo da iluminação de cameras de vídeo (Sanyo) [REZNIK].

2.3. CONJUNTOS DIFUSOS E LÓGICA DIFUSA: A BASE DO CONTROLO DIFUSO

2.3.1. CONJUNTOS CLÁSSICOS

O conceito de conjunto é bastante antigo, análogo ao conceito de um ponto ou de uma recta. Daqui para adiante um conjunto é definido por algo que pode conter ou não elementos, sendo estes elementos definidos por objectos. Os objectos, se existirem, formam um conjunto de elementos.

Na teoria dos conjuntos trabalha-se normalmente com vários conjuntos, cujos elementos são todos do mesmo tipo, pertencem todos a um mesmo aglomerado, denominado conjunto universo.

Se A é um conjunto, a afirmação “ x é um elemento de A ” representa-se por $x \in A$, e pelo contrário, “ x não é um elemento de A ” representa-se por $x \notin A$.

Por exemplo, se A é o conjunto dos três primeiros números naturais pares, então $2 \in A$, $4 \in A$, $6 \in A$, $24 \notin A$, $9 \notin A$, são afirmações verdadeiras. Desta forma, pode-se afirmar que a “pertença” é uma relação que vincula cada elemento a um conjunto e, não entre elementos de um conjunto. Esta afirmação será muito útil para compreender a definição das funções de pertença dos conjuntos difusos [MIRANDA].

2.3.1.1. FUNÇÕES DE PERTENÇA

Como foi mencionado acima, a relação de pertença é algo que associa cada elemento a um conjunto. Por outras palavras, num conjunto bem definido (lógica clássica), a pertença ou não pertença de um elemento x a um conjunto A descreve-se conforme a função característica $\mu_A(x)$, em que:

$$\mu_A(x) = \begin{cases} 1, & x \in A \\ 0, & x \notin A \end{cases}$$

Esta função é chamada função de pertença ou função característica de A e está definida para todos os elementos do universo. A função de pertença procura em todo o universo U os dois elementos $\{0,1\}$ num determinado conjunto:

$$\mu_A: U \rightarrow \{0,1\}$$

Definida a função característica como $\{0,1\}$ ou {falso, verdadeiro}, poder-se-á atribuir valores verdadeiros a proposições referentes ao conjunto A [MIRANDA].

2.3.1.2. OPERAÇÕES COM CONJUNTOS

Pode-se construir os conjuntos a partir de dois ou mais conjuntos dados, aplicando-lhes as operações básicas que nos são bem familiares: União, Intersecção e a Negação (complemento). Normalmente, para representar os universos U , os conjuntos e os elementos aplicam-se os diagramas de Venn. Na figura 1 está ilustrada a representação desses mesmos diagramas. [MIRANDA]

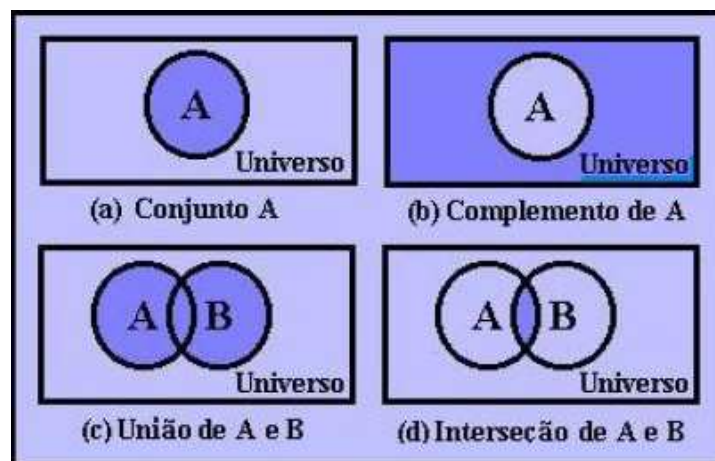


Figura 1 - Representação de operações de conjuntos utilizando Diagramas de Venn [MIRANDA]

2.3.2. CONJUNTOS DIFUSOS

Toda a lógica usada no controlo difuso tem por base os conjuntos difusos. Estes estão definidos por funções de pertença, as quais retornam a veracidade de uma variável. Por outras palavras, indicam-nos se uma dada variável pertence ou não a um conjunto.

Um conjunto difuso D , de um universo $U = \{x|x \in \mathfrak{R}\}$, define-se como uma comparação $\mu_D(x): U \rightarrow [0, \alpha]$, $\alpha \in \mathfrak{R}$, no qual a cada x é atribuído um número no intervalo entre $[0, \alpha]$ o que nos indica o valor de pertença de x ao conjunto difuso D . Por exemplo, se U é o conjunto das alturas possíveis para o ser humano e $D = \text{médio}$, $\mu_{\text{medio}}(x)$ indica-nos quanto x pertence a D .

Quando se generaliza a função de pertença ($\alpha = 1$), obtém-se $\mu_D(x): U \rightarrow [0,1]$. Deste ponto do relatório e até ao seu fim será utilizada sempre a função de pertença geral. Para os casos extremos, a função de pertença pode reduzir-se a singularidades, ou seja, a lógica difusa passa a ser lógica clássica. Se as singularidades podem apenas ter dois estados possíveis, 0 ou 1, então estamos perante lógica binária.

Por exemplo: para se obter os valores normalizados de um conjunto de números, divide-se cada número pelo número com maior valor, assim a divisão do maior número por ele mesmo resultará no valor 1. Seja o conjunto $\{30, 50, 80, 100, 70, 40\}$, o qual quando normalizado resulta em $\{0.3, 0.5, 0.8, 1, 0.7, 0.4\}$. Para um conjunto difuso o conjunto de avaliação é um intervalo real:

$$\mu_D(x): U \rightarrow [0,1]$$

A função de pertença de um conjunto difuso permite uma vasta continuidade de graus de pertença. Assim, em vez dos dois estados apenas, 0 e 1 ou verdadeiro e falso, conseguem-se infinitos valores para o grau de pertença.

O grau de pertença não representa uma probabilidade, mas sim uma medida de comparação de um objecto com um determinado conjunto difuso [MIRANDA].

Um exemplo de uma função de pertença é ilustrado na Figura 2, onde:

$$\mu_P(x) = \begin{cases} 1, & \text{se } x < 1.5 \\ 2.5 - x, & \text{se } 1.5 \leq x \leq 2.5 \\ 0, & \text{se } x > 2.5 \end{cases}$$

$$\mu_M(x) = \begin{cases} x - 1.5, & \text{se } 1.5 \leq x \leq 2.5 \\ 3.5 - x, & \text{se } 2.5 \leq x \leq 3.5 \\ 0, & \text{se } x < 1.5 \text{ ou } x > 3.5 \end{cases}$$

$$\mu_G(x) = \begin{cases} 0, & \text{se } x < 2.5 \\ x - 2.5, & \text{se } 2.5 \leq x \leq 3.5 \\ 1, & \text{se } x > 3.5 \end{cases}$$

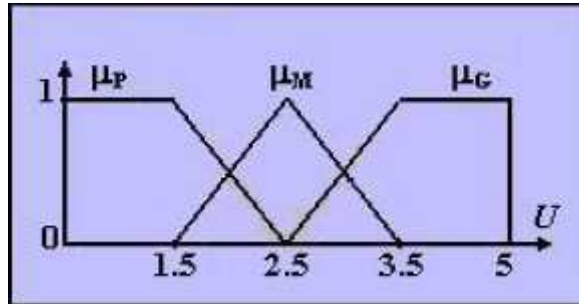


Figura 2 - Funções de Pertença para valores pequenos, médios e grandes [MIRANDA]

2.3.3. OPERAÇÕES COM CONJUNTOS DIFUSOS

Na lógica booleana a função dos operadores são bem conhecidas, se por exemplo aplicarmos a operação *And* temos:

$$1 \text{ And } 1 \rightarrow 1$$

$$1 \text{ And } 0 \rightarrow 0$$

$$0 \text{ and } 1 \rightarrow 0$$

$$0 \text{ And } 0 \rightarrow 0$$

Ou seja, se o elemento pertence a dois conjuntos, então ele próprio pertence à intersecção. Se pertence a um e não ao outro não pertence à intersecção.

Existindo dois conjuntos difusos A e B e se lhes for aplicada uma operação de união (ou *Or*), a pertença dos elementos ao conjunto união é obtida tomando-se o maior valor de pertença relativo aos dois conjuntos. Vejamos, se $\mu_A(x) = 0.5$ e $\mu_B(x) = 0.7$ então $\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\} = \max\{0.5, 0.7\} = 0.7$. A pertença do valor x à união difusa de A e de B é 0.7.

Para o caso da intersecção (ou *And*), o valor resultante da operação corresponde ao valor mínimo da pertença dos dois conjuntos. Considerando o exemplo acima, $\mu_{A \cap B}(x) =$

$\min\{\mu_A(x), \mu_B(x)\} = \min\{0.5, 0.7\} = 0.5$. A pertença do valor x à intersecção difusa de A e de B é 0.5.

Na operação negação (ou *Not*), considera-se o valor que somado ao elemento resulta no valor 1. Considerando o exemplo anterior, a pertença de x à negação do conjunto difuso A, é dada por $\mu_{\bar{A}}(x) = 1 - \mu_A(x)$ [MIRANDA].

2.3.4. UNIÃO DE DOIS CONJUNTOS DIFUSOS

Sejam A e B dois conjuntos difusos no universo U com funções de pertença $\mu_A(x)$ e $\mu_B(x)$, respectivamente. A união dos conjuntos A e B é dada por:

$$A \cup B = \{(\frac{\max\{\mu_A(x), \mu_B(x)\}}{x}) | x \in U\}$$

A função de pertença de x ao conjunto de união entre A e B é:

$$\mu_{A \cup B}(x) = \max\{\mu_A(x), \mu_B(x)\}$$

na qual a função $\max\{\}$ selecciona o valor máximo dos graus de pertença dos dois conjuntos difusos. Na Figura 3 é ilustrado um exemplo da união de dois conjuntos difusos [MIRANDA].

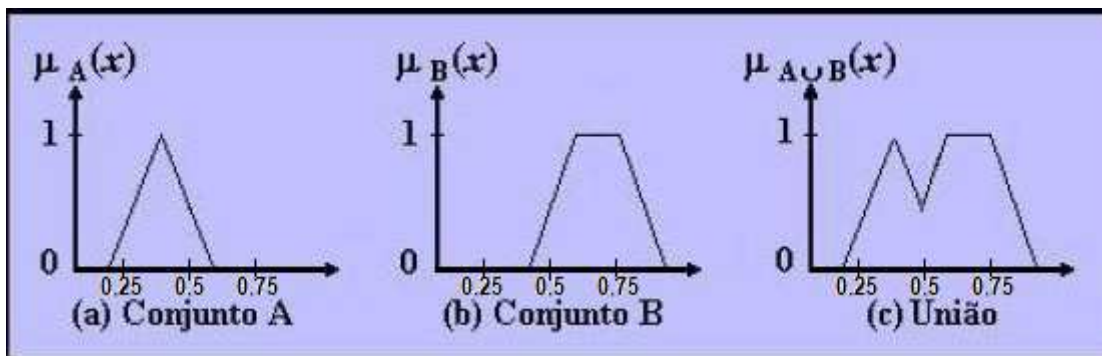


Figura 3 - Operação de União entre dois Conjuntos Difusos A e B [MIRANDA]

2.3.5. INTERSECÇÃO DE DOIS CONJUNTOS DIFUSOS

Seja A e B dois conjuntos difusos no universo U com funções de pertença $\mu_A(x)$ e $\mu_B(x)$ respectivamente, a intersecção de ambos os conjuntos define-se por:

$$A \cap B = \{(\frac{\min\{\mu_A(x), \mu_B(x)\}}{x}) | x \in U\}$$

A função de pertença de x ao conjunto de intersecção entre A e B é:

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\}$$

na qual a função $\min\{\}$ selecciona o valor mínimo dos graus de pertença. A Figura 4 ilustra um exemplo de intersecção entre dois conjuntos difusos [MIRANDA].

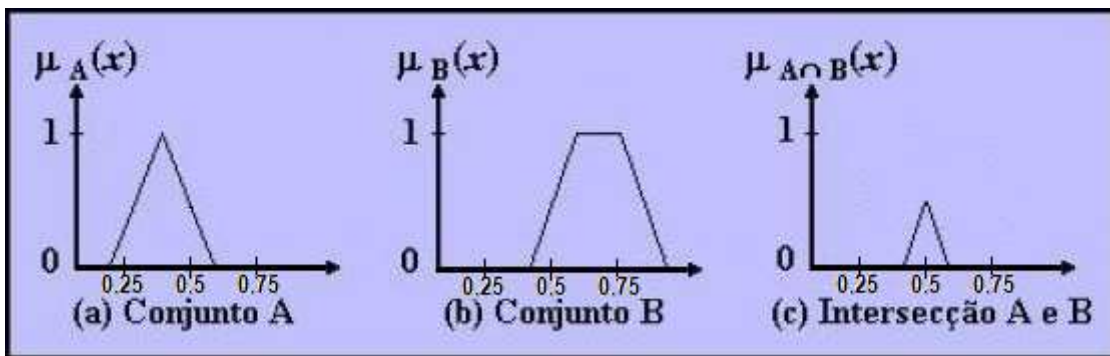


Figura 4 - Operação de Intersecção entre dois Conjuntos Difusos A e B [MIRANDA]

2.3.6. NEGAÇÃO DE UM CONJUNTO DIFUSO

Seja A um conjunto difuso no universo U com uma função de pertença $\mu_A(x)$, a negação do conjunto A é o conjunto \bar{A} (A negado) definido por:

$$\bar{A} = \{(\frac{1 - \mu_A(x)}{x}) | x \in U\}$$

A função de pertença de x ao conjunto \bar{A} é:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)$$

A Figura 5 ilustra dois exemplos de negação de um conjunto difuso [MIRANDA].

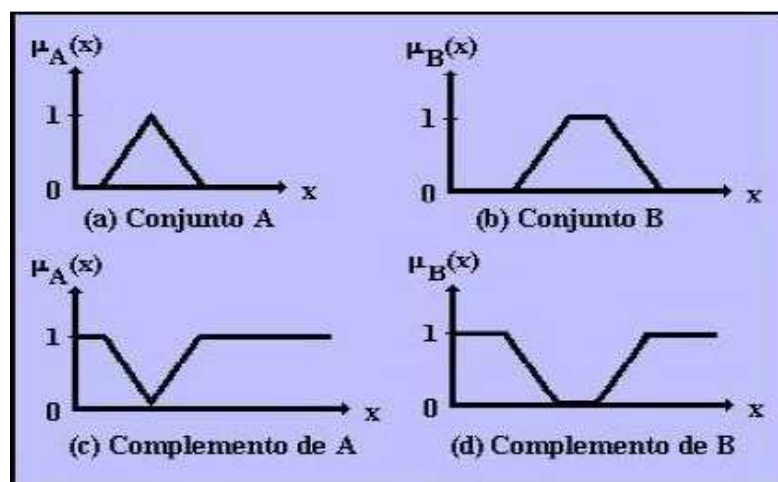


Figura 5 - Operação de Negação de dois Conjuntos Difusos A e B [MIRANDA]

2.4. LÓGICA DIFUSA

A lógica difusa é usada para fazer representações difusas do conhecimento. Quando um ser humano está a tentar resolver um problema complexo, primeiro estrutura todo o conhecimento que se têm sobre o problema em conceitos gerais e depois, observa as relações entre esses conceitos. Este processo de modelação *top-down* permite que se convertam relações essencialmente gerais e imprecisas em algoritmos operacionais mais detalhados. Esta perspectiva humana de resolver um problema, geralmente não permite a definição precisa de uma solução em termos de números exactos, mas conduz a uma classificação ou agregação qualitativa em categorias gerais ou conjuntos de possíveis soluções. A capacidade de classificar de modo impreciso as variáveis de um problema, em termos de conceitos qualitativos em vez de quantitativos, traduz a ideia de uma variável linguística. O processo de representação difuso do conhecimento aqui descrito, depende fundamentalmente deste conceito [MIRANDA].

2.4.1. VARIÁVEIS LINGUÍSTICAS

Uma variável linguística é definida como uma entidade utilizada para representar, de modo impreciso e portanto linguístico, uma variável de um dado problema. Para definir uma variável linguística utilizam-se apenas expressões linguísticas como frio, muito grande, aproximadamente alto, etc. Estes valores contrastam com os valores assumidos por uma variável numérica, que admite apenas valores precisos e concretos, ou seja, números.

Um termo primário (frio, pequeno, calor, etc) de uma dada variável linguística pode ser representado por um conjunto difuso existente num universo, no qual esta variável está definida. Assim, cada conjunto difuso definido neste universo é associado a um conceito linguístico que classifica ou define um valor impreciso para a variável em questão. Para um dado elemento x de um universo, o valor de pertença $\mu_D(x)$ representa o quanto este elemento pertence ao conjunto difuso D .

Os termos primários definidos para uma dada variável linguística formam a sua estrutura de conhecimento, denominada de conjunto de funções de pertença dessa mesma variável. Na Figura 6 é ilustrado um exemplo de um conjunto difuso de funções de pertença que representam a variável linguística Temperatura e, o universo compreende-se numa escala entre 0 a 50 graus Celsius.

A forma como se utilizam as variáveis linguísticas, depende basicamente da definição das propriedades sintáticas e semânticas que vão reger o comportamento do sistema de conhecimento difuso. As propriedades sintáticas definem o formato em que serão armazenadas as informações linguísticas difusas. Elas proporcionam a criação de uma base de conhecimento contendo frases estruturadas, sintetizando os processos de armazenamento, procura e processamento dos dados existentes [MIRANDA].

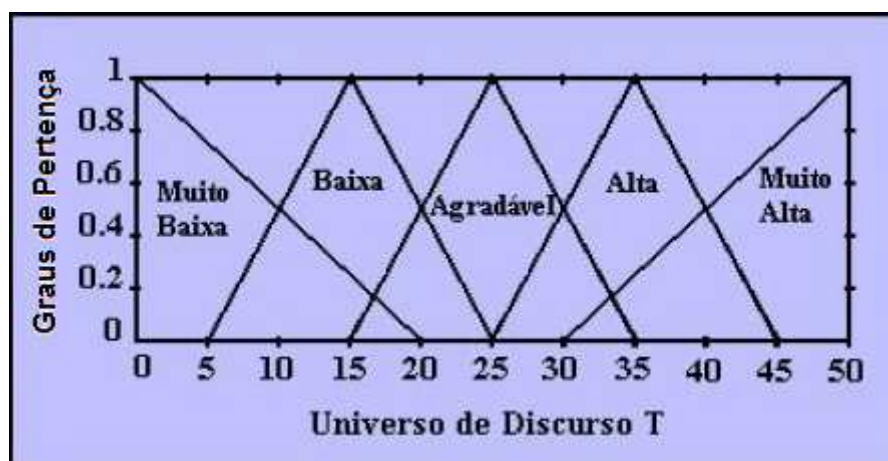


Figura 6 - Conjunto de funções de pertença da variável Temperatura [MIRANDA]

Por outro lado, as propriedades semânticas vão especificar de que modo é captado e processado o conhecimento, armazenado na forma de declarações condicionais difusas, ou em regras de produção difusas contidas na estrutura definida pelas propriedades sintáticas.

Uma variável linguística, como o seu próprio nome indica, é uma variável cujos valores são palavras ou frases numa linguagem natural ou sintética. Na expressão “a velocidade de um carro”, o termo Velocidade é uma variável linguística e os seus valores são: muito baixa, baixa, média, alta, muito alta e, assim sucessivamente, conforme é ilustrado na Figura 7 [MIRANDA].

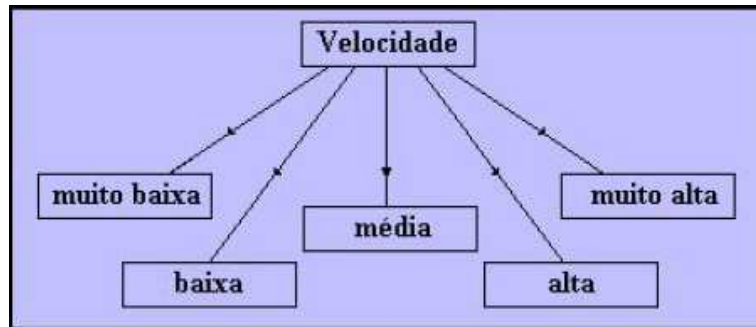


Figura 7 – Valores linguísticos da variável Velocidade [MIRANDA]

Cada valor de uma variável linguística representa um conjunto difuso num determinado universo. Concretamente, uma variável linguística é formada por cinco partes $(x, T(x), U, G, M)$, em que [MIRANDA]:

- x : nome da variável;
- $T(x)$: conjunto de valores linguísticos de x ;
- U : universo de discurso em que se define $T(x)$;
- G : regra sintática para gerar os nomes dos valores de x ;
- M : regra semântica para associar cada valor ao seu significado.

Por exemplo: Velocidade é uma variável linguística. O conjunto de valores linguísticos (partição difusa do seu universo) é:

$$T(\text{velocidade}) = \{\text{muito baixa}, \text{baixa}, \text{média}, \text{alta}, \text{muito alta}\}$$

Cada termo em $T(\text{velocidade})$ está caracterizado por um conjunto difuso no universo de discurso, exemplo: $U=[0,200]$ km/h. A regra sintática G determina a ordem das palavras dos valores linguísticos de Velocidade: como em muito alta, na qual muito é um modificador que precede ao termo primário alta. A regra semântica M associa cada valor linguístico com o seu significado: {alta é maior próxima de 180}, e {baixa é menor próxima de 30}, etc.

Supondo agora outro caso, o pêndulo invertido. As variáveis linguísticas são: ângulo, velocidade angular e força. Os valores linguísticos para cada variável difusa são: NG

(negativo grande), NM (negativo médio), NF (negativo fraco), ZE (zero), PF (positivo fraco), PM (positivo médio) e PG (positivo grande), conforme ilustra a Figura 8 [MIRANDA].

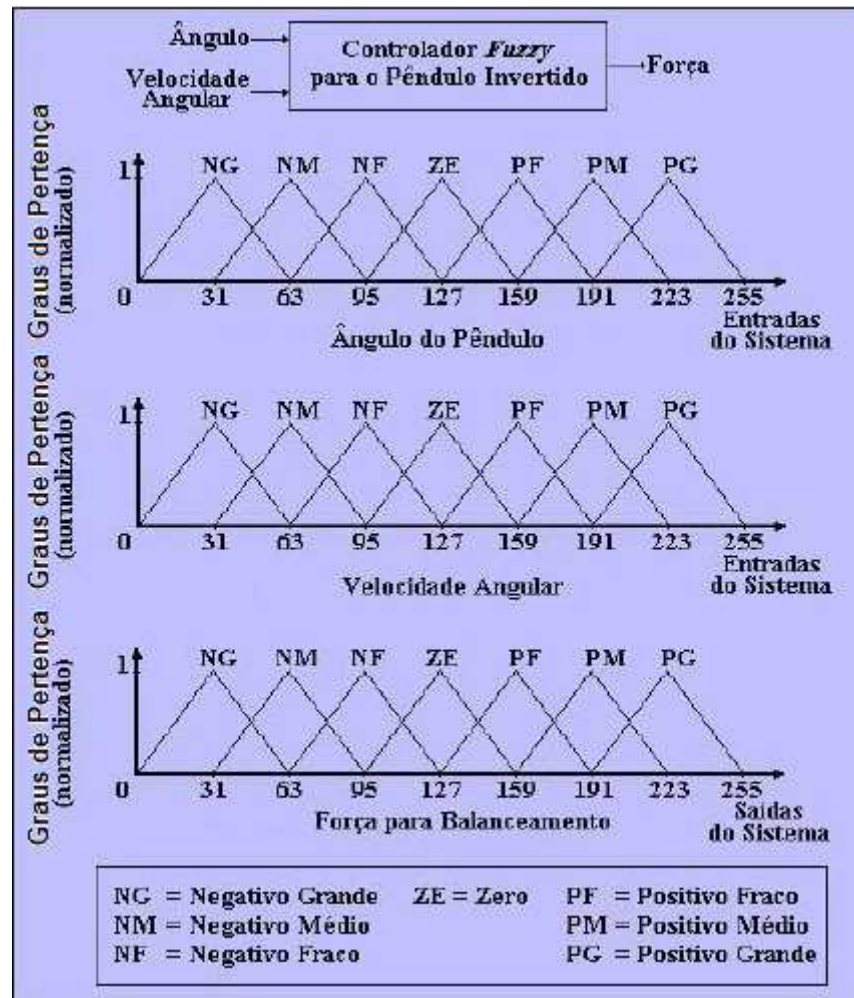


Figura 8 - Funções de pertinência do pêndulo invertido [MIRANDA]

2.4.2. REGRAS DE PRODUÇÃO DIFUSA

A maneira mais usual de armazenar informações numa base de conhecimento difusa é através da representação por meio de regras difusas. Uma regra difusão é normalmente formada por duas partes principais:

$$\text{If } \langle \text{condição} \rangle \text{ Then } \langle \text{consequência} \rangle$$

O campo *condição* é onde são descritas as condições que quando satisfeitas (mesmo que parcialmente), determinam as consequências através dum mecanismo de inferência difusa. A este processo denomina-se activação de uma regra.

Por sua vez, a *consequência* é composta por um conjunto de acções ou diagnósticos que são gerados com a activação de uma regra. Todas as consequências são geradas em conjunto para produzir uma resposta de saída para cada variável do sistema [MIRANDA].

2.4.3. DEFINIÇÃO DAS REGRAS DE CONTROLO

As regras de controlo englobam o conhecimento do sistema e os objectivos do controlo. Cada regra tem definido todos os possíveis estados do sistema na sua entrada e uma acção de controlo correspondente na sua saída. As regras de controlo difusas ligam os valores da entrada com as propriedades de saída do sistema. Estas são expressas como proposições condicionais:

Se [Estado do processo] Então [Acção de Controlo]

Como exemplo de regras de controlo temos [MIRANDA]:

Se Temperatura é fria E Pressão é alta Então Acção da válvula é positiva média

2.4.4. ORDEM DAS REGRAS

As regras de controlo difusas são declarativas, e não sequenciais, o que significa que a ordem em que se expressam não é importante. Como uma medida preventiva para a manutenção do controlador é recomendável agrupar as regras de acordo com as suas condições [MIRANDA].

Se Temperatura é fria E Pressão é alta Então Acção da válvula é positiva pequena

Se Temperatura é fria E Pressão é média Então Acção da válvula é positiva normal

Se Temperatura é fria E Pressão é baixa Então Acção da válvula é positiva grande

Se Temperatura é média E Pressão é alta Então Acção da válvula é não operar

Se Temperatura é média E Pressão é média Então Acção da válvula é não operar

Se Temperatura é média E Pressão é baixa Então Acção da válvula é positiva

2.4.5. NÚMERO DE REGRAS

O número de regras necessárias para definir um controlador é obtido através da multiplicação do número de termos difusos por eles mesmos. Deste modo, as regras cobrirão todas as possíveis combinações provenientes das distintas entradas. Por exemplo, para um sistema com duas variáveis de entrada, cada uma com 5 termos difusos, existem $5 \times 5 = 25$ combinações possíveis de entrada e portanto 25 regras de controlo [MIRANDA].

2.4.6. DESENHO DE UMA BASE DE REGRAS

Para construir uma base de regras utiliza-se uma matriz que cubra todas as possíveis combinações das entradas. Para um sistema com duas entradas, atribui-se uma para cada eixo da matriz. Em cada célula da matriz, escreve-se a acção de controlo sugerida pela regra que defina esta combinação de entrada como condição. Da matriz pode-se obter todas as regras que formarão uma base de regras completa. Se o sistema tiver três entradas utiliza-se uma matriz para cada termo linguístico da terceira variável [MIRANDA].

Para o caso do pêndulo invertido, tem-se a seguinte matriz:

		Ângulo						
		NG	NM	NF	ZE	PF	PM	PG
Velocidade Angular	NG				NG			
	NM				NM			
	NF				NF	PF		
	ZE	NG	NM	NF	ZE	PF	PM	PG
	PF			PF	PF			
	PM				PM			
	PG				PG			
		Se Ângulo é NG E Velocidade Angular é ZE Então Força é NG						

Figura 9 - Matriz de regras para o pêndulo invertido [MIRANDA]

Em alguns casos, é possível utilizar menos regras, mas não é recomendável fazê-lo, pois elas representam conhecimento sobre o sistema a controlar. Se por ventura, alguma regra

for eliminada, remove-se conhecimento o qual pode tornar-se importante se o sistema for modificado posteriormente [MIRANDA].

2.5. ARQUITECTURA DE UM SISTEMA DIFUSO

O Controlo Lógico Difuso fornece uma metodologia para representar, manipular e implementar o conhecimento heurístico humano com vista a controlar qualquer tipo de processo. A Figura 10 ilustra um diagrama de blocos com a estrutura interna de um controlador difuso embebido num sistema em malha fechada [KEVIN].

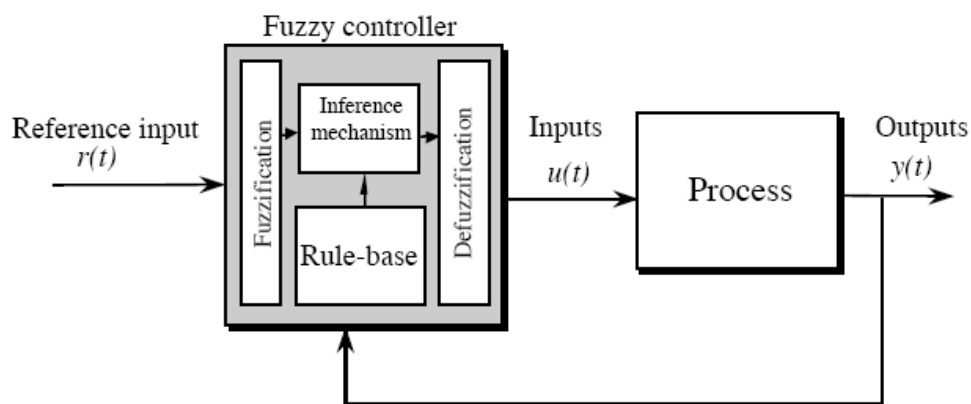


Figura 10 - Arquitectura de um Controlador Lógico Difuso [KEVIN]

O Controlador Lógico Difuso é composto por quatro componentes principais: (1) A base de regras (*Rule-Base*) onde se coloca o conhecimento de controlo na forma de regras que define a melhor forma de controlar um sistema. (2) O mecanismo de inferência (*inference mechanism*) que avalia quais as regras de controlo que estão activas num determinado período de tempo e atribui um valor correspondente à saída do próprio bloco. (3) O bloco de difusão (*fuzzification*) converte os sinais das entradas de forma a que os valores introduzidos possam ser interpretados e comparados às regras definidas na base de regras e, por último, o bloco de colapsagem (*defuzzification*) que converte os resultados obtidos no mecanismo de inferência em valores de entrada do sistema a controlar.

Basicamente, um controlador com lógica difusa é visto como um “decisor” artificial que opera num sistema em malha fechada e em tempo real. Para programar um controlador lógico difuso, o programador necessita de reunir informações básicas de como o controlador se deve comportar para controlar um determinado processo, e colocar essa

informação sob a forma de regras. Normalmente, esta mesma informação poderá ser dada por um operador que normalmente executa a tarefa de controlo manualmente [KEVIN].

2.5.1. DIFUSÃO

O processo de difusão atribui ou calcula o grau de pertença de uma entrada, que pertença a um ou mais conjuntos difusos. A Figura 11 ilustra uma entrada do sistema, *Temperatura*, contendo quatro conjuntos difusos: *fria*, *fresca*, *morna* e *quente*. Cada valor que a variável *Temperatura* pode assumir, tem um grau de pertença associado a cada um dos quatro conjuntos difusos. Este mesmo grau de pertença é definido com base na experiência e no conhecimento que o utilizador tem sobre o sistema.

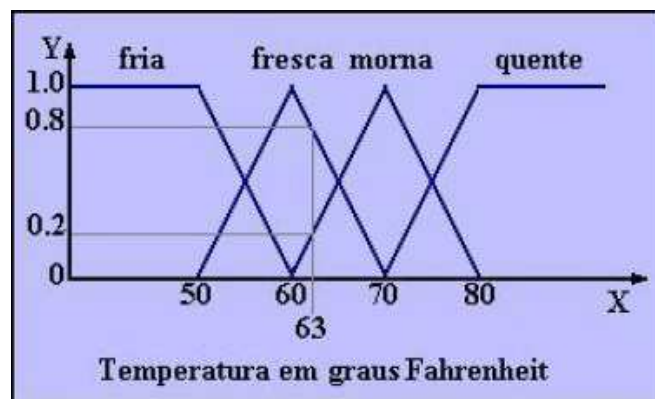


Figura 11 - Conjuntos Difusos da variável Temperatura [MIRANDA]

Um quinto conjunto difuso poderia ser adicionado aos quatro existentes, chamando-se *confortável* e ficaria localizado entre *fresca* e *morna*. Contudo, o número de conjuntos difusos varia conforme as características de controlo que se pretende para o sistema, nomeadamente: precisão da resposta, estabilidade, facilidade de implementação e manipulação de dados. Estas são algumas das mais importantes características a ter em conta quando se definem as funções de pertença.

Para cada entrada do sistema, o conjunto difuso deve abranger o eixo inteiro dos XX. Para o eixo dos YY o seu valor varia entre 0 e 1. A sobreposição de limites é bastante importante para o funcionamento do sistema, pois permite a pertença de um valor em vários conjuntos, mesmo que aparentemente contraditórios. Por exemplo, na Figura 11, para uma entrada de temperatura de 63° F, esta pode corresponder simultaneamente a *fresco* e *morno*, mas com um maior valor de pertença para *fresco*. Trata-se de uma boa

prática, permitir uma sobreposição de 25% entre os conjuntos difusos, pois fornece uma relação contínua entre o valor de entrada e a respectiva variável linguística.

O principal objectivo do processo de difusão é criar uma forte correspondência entre os termos linguísticos (frio, aproximadamente, activo, grande) e as funções de pertença, permitindo ao programador expressar ou modificar o comportamento de um sistema de uma forma clara e concisa. Esta propriedade torna as tarefas complexas mais simples de definir. Outra propriedade importante é a imutabilidade das funções de pertença quando o sistema está em funcionamento [MIRANDA].

Até agora a forma física para definir as funções de pertença, foi a triangular no entanto, existem outras formas que são ilustradas na Figura 12.

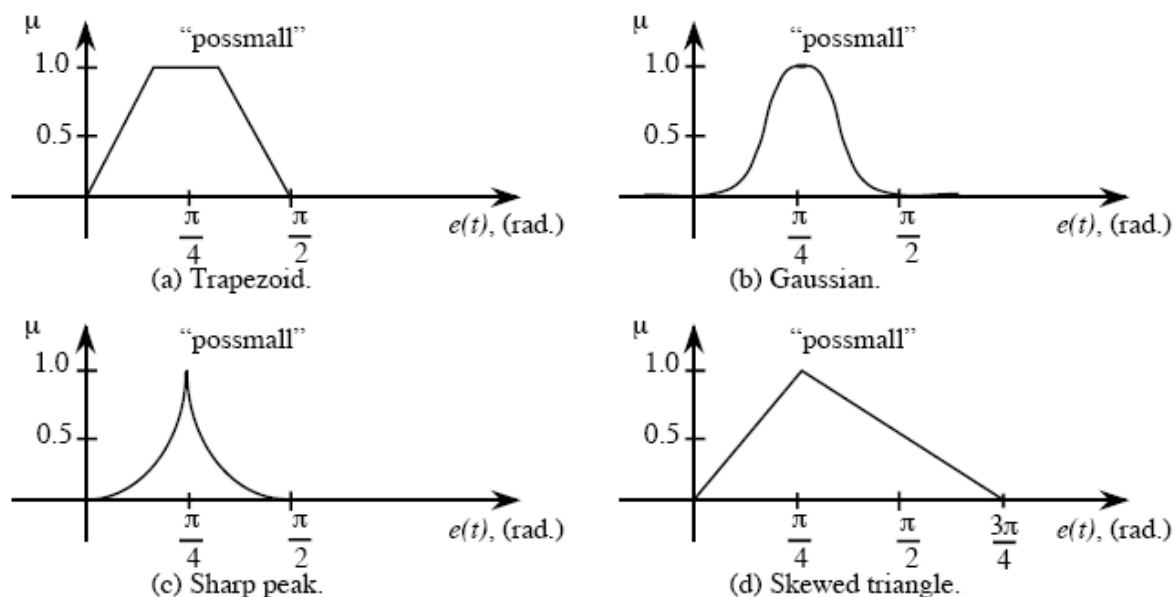


Figura 12 - Tipos de Funções de Pertença. (a) trapezoidal, (b) gaussiana, (c) pico, (d) triangular
[KEVIN]

A forma para a função de pertença mais usada é a triangular, a qual oferece melhores resultados na maioria das aplicações, contudo qualquer forma pode ser adaptada a diversas situações [KEVIN]. Para uma selecção de qual a forma a utilizar, poder-se-á consultar o livro "Fuzzy Control" de Kevin Passino.

2.5.2. CÁLCULO DO GRAU DE PERTENÇA

Para calcular o grau de pertença é necessário determinar os termos *delta1* e *delta2*, conforme a Figura 13.

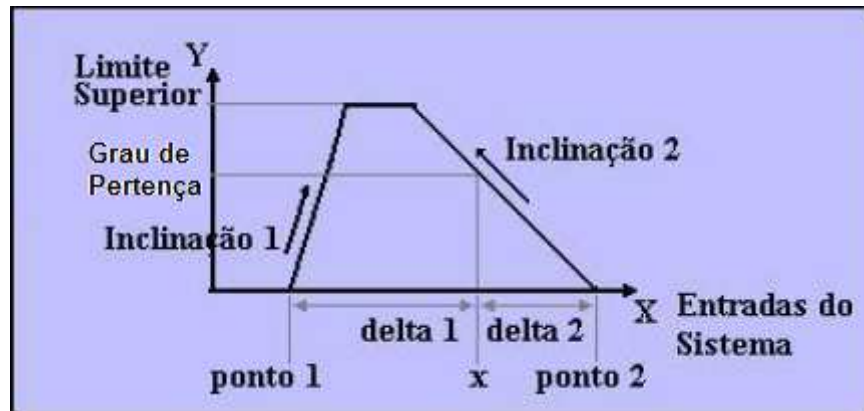


Figura 13 - Método de Cálculo dos Graus de Pertença [MIRANDA]

A determinação do grau de pertença, como o próprio nome indica, permite saber se uma dada entrada pertence ou não ao conjunto difuso, caso pertença este retorna o seu grau de pertença. Por exemplo:

$$\text{delta1} = (x - \text{ponto1})$$

$$\text{delta2} = (\text{ponto2} - x)$$

Se

$$\text{delta1} \leq 0 \text{ ou } \text{delta2} \leq 0 \rightarrow \text{Então Grau de Pertença} = 0$$

Senão

$$\text{Grau de pertença} = \min \{ \text{delta1} * \text{declive1}, \text{delta2} * \text{declive2}, \text{limite superior} \}$$

NOTA: O cálculo do declive é obtido através da divisão do valor em YY pelo valor em XX.

2.5.3. BASE DE REGRAS OU BASE DO CONHECIMENTO

Para controlar o comportamento de um sistema, o programador tem obrigatoriamente de criar uma série de regras na forma Se-Então. Estas regras serão a base do controlo.

Considerando o exemplo de um mercado de acções, conforme a Figura 14, temos:

Se Preço das acções está a decrescer e o Volume de negócios é grande → Então a Ordem é vender.

Os termos *Preço das acções está a decrescer* e *o Volume de negócios é grande* são as condições, que têm associado um grau de pertença como resultado da difusão. Neste caso é *vender acções* [MIRANDA].

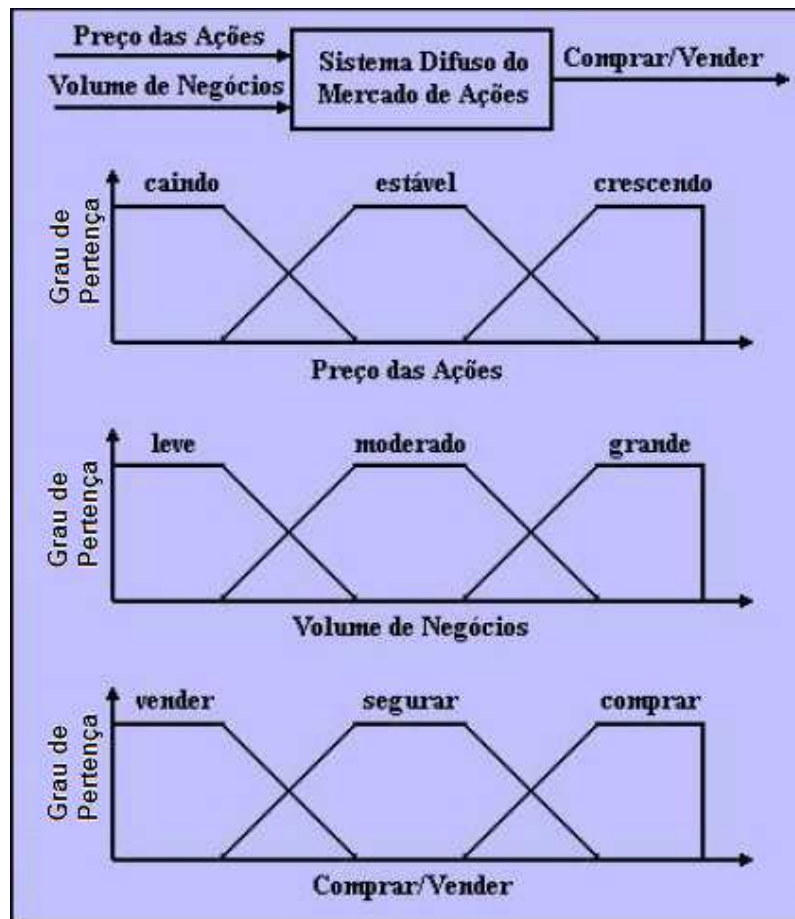


Figura 14 - Sistema Difuso de mercado de acções [MIRANDA]

Outro exemplo, bastante comum, de aplicação de regras difusas é o caso de uma caldeira, na qual existe uma relação entre a *Temperatura* interna e o *Caudal* de água necessário para a arrefecer, o que produz as seguintes regras:

Se Temperatura é alta Então Caudal é alto;

Se Temperatura é baixa então Caudal é baixo;

Se Temperatura é ideal Então Caudal é normal.

De uma forma simples, a temperatura da caldeira mantêm-se estável ou constante, desde que os conjuntos difusos tenham sido definidos de acordo com a realidade.

Pode-se assim concluir neste tópico que o desenvolvimento das regras de controlo tem de levar em conta o seguinte: Primeiro, a escolha do conjunto de variáveis linguísticas que descrevam apropriadamente os parâmetros de controlo do processo; Segundo, a escolha da gama de valores para cada entrada e saída é vital para a suavidade de resposta de controlo; Terceiro, e último, a base de regras ou conhecimento deverão ser construídos consoante o significado dos parâmetros linguísticos [MIRANDA].

2.5.4. COLAPSAGEM

Como o próprio termo em inglês indica, *Defuzzification* trata do processo inverso à difusão (*Fuzzification*), ou seja, converte informações qualitativas em informação quantitativa. Os métodos mais utilizados para esta tarefa são: o método por Centro de Área e o método da Média dos Máximos. Os quais são descritos a seguir.

Método do Centro de Área – COA

Um dos métodos mais utilizados para o processo de colapsagem é o método do Centro de Área, que também é conhecido como Centro de Gravidade ou Centróide.

Este método processa-se da seguinte forma. Um ponto central no eixo dos XX é determinado para cada função de pertença de saída, as quais são limitadas em altura pela força da regra aplicada e assim obtêm-se as áreas das funções de pertença. Por último, a saída sofre a colapsagem que provém de uma média ponderada dos pontos centrais, ou centróides, do eixo dos XX e as áreas calculadas, com as áreas a servir como peso. Este método é ilustrado na Figura 15. O método Centro de Área calcula a abcissa do ponto do centro de área e utiliza-a como valor escalar de saída para um determinado conjunto difuso de saída [MIRANDA].

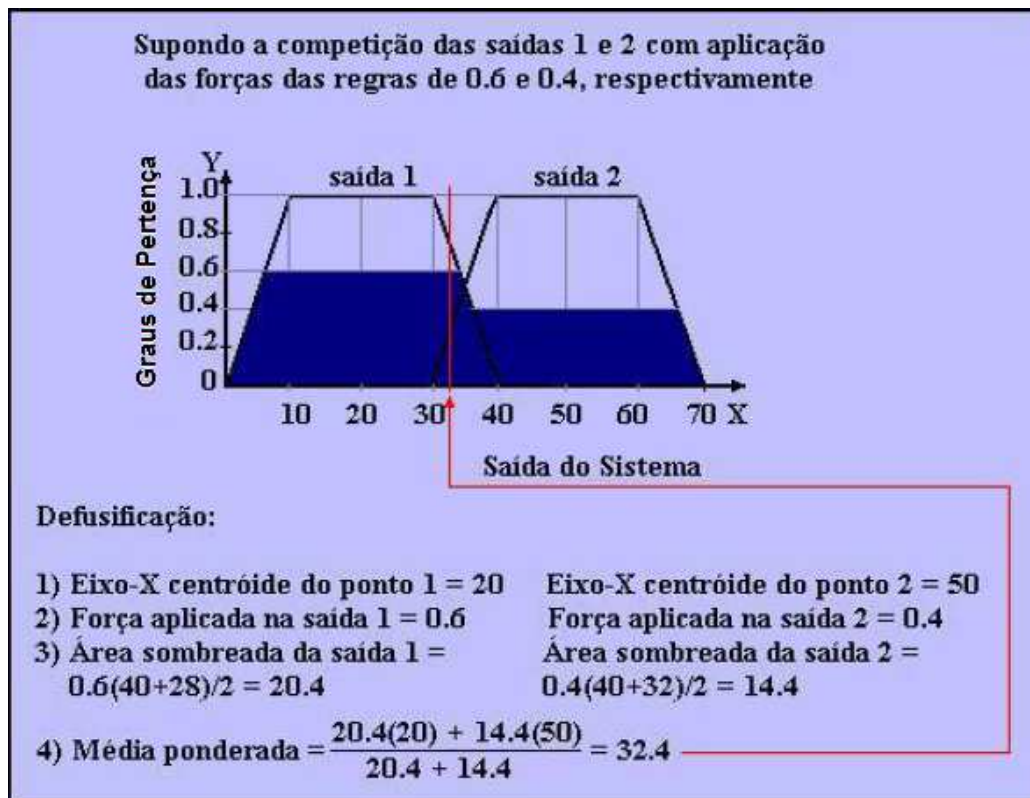


Figura 15 - Colapsagem - Método de Centro de Área [MIRANDA]

Estando uma acção de controlo, relacionada com uma função de pertinência dada por $\mu_C(z)$, o método do Centro de Área calcula o centro de gravidade para a acção de controlo utilizando a seguinte expressão:

$$Z^* = \frac{\sum_{j=1}^q Z_j \mu_C(Z_j)}{\sum_{j=1}^q \mu_C(Z_j)}$$

em que o termo q corresponde ao número de níveis de quantificação da saída, ou seja, é o número de regras cujo valor da função de pertinência é superior a zero, Z_j é a soma das saídas de controlo no nível de quantificação j , e $\mu_C(Z_j)$ representa o valor da função de pertinência em C .

Método da Média do Máximos – MOM

Para este método o valor para a acção de controlo é obtido através do ponto médio de todos os valores de pertinência máximos, ou seja calcula a média de todos os valores que alcançam o mesmo máximo na saída difusa final, conforme ilustra a Figura 16.



Figura 16 - Colapsagem - Método da Média dos Máximos MOM [MIRANDA]

Este método baseia-se na seguinte expressão:

$$Z^* = \sum_{j=1}^l \frac{Z_j}{l}$$

em que l representa o número de valores quantificados, cuja função de pertença é máxima [MIRANDA].

2.5.5. MÁQUINA DE INFERÊNCIA

O mecanismo de inferência tem duas tarefas básicas. A primeira, é quantificar o quanto cada regra é relevante para a situação actual e caracteriza-la pelas entradas $u_i, i = 1, 2, \dots, n$; A segunda é desenhar a saída do sistema usando as entradas em questão u_i e a informação da base de regras [KEVIN], ver Figura 17.



Figura 17 – Diagrama típico de um sistema de processamento difuso [MIRANDA]

2.6. CONCLUSÕES

Pode-se concluir com este capítulo que a teoria sobre a Lógica Difusa surge como uma abordagem diferente ao controlo de sistemas. Este método foca-se principalmente no comportamento de cada sistema, em particular tentando que este se comporte de acordo com o conhecimento heurístico que o programador tem sobre o mesmo, ao contrário de outros métodos de sintonia nos quais o primeiro passo para o controlo é a obtenção do modelo matemático do sistema.

Contudo, este tipo de controlo exige algum conhecimento teórico para se definir correctamente o conjunto de regras difusas, a gama de valores de entrada e saída do controlador e a forma como os diversos graus de pertença se interrelacionam entre si.

Conclui-se assim que, quanto maior for o grau de complexidade do sistema a controlar maior será o nível de dificuldade para parametrizar o controlador difuso.

3. ANÁLISE DA RESPOSTA DE SISTEMAS

Neste capítulo são abordados conceitos teóricos que descrevem o comportamento de sistemas quando sujeitos a uma determinada entrada. Os sistemas em estudo são de primeira, segunda e de ordens superiores.

3.1. INTRODUÇÃO

Normalmente, nos sistemas físicos reais o sinal de entrada do próprio sistema é desconhecido, aleatório e os seus valores instantâneos não podem ser expressos de maneira analítica [OGATA]. Contudo, para a análise e projecto de um determinado sistema de controlo deve-se adoptar alguns procedimentos padrão para comparar o seu desempenho, ou seja, estimulando-se os sistemas com sinais de entrada conhecidos e comparando as respostas de vários sistemas a estes sinais.

Os sinais de entrada típicos que geralmente são usados para estimular os sistemas são: entrada em degrau, em rampa, parábola de aceleração, impulso, sinusoidais e outras. Com estes tipos de sinais de entrada consegue-se facilmente uma análise experimental e matemática dos sistemas de controlo, pois os próprios sinais de entrada são funções em ordem ao tempo, muito simples. A escolha do tipo de entrada é feita mediante a

observação do sistema real e a observação do tipo de sinal de entrada que este estará sujeito com maior frequência sob as condições normais de operação [HELIO]. Se a entrada de um sistema de controlo é função do tempo e este varia gradualmente, então a entrada em rampa será um bom sinal de teste. Da mesma forma que, um sistema que seja submetido a variações bruscas de entrada, um sinal de entrada em degrau será um bom sinal de teste, por último se um sistema estiver sujeito constantemente a entradas do tipo impacto, logo um sinal de entrada em impulso será a melhor forma de testar o sistema em causa [OGATA].

Logo que a escolha do tipo de entrada seja feita, supondo que seja a mais adequada, o desempenho do sistema em resposta à entrada real geralmente é satisfatório. O uso destes sinais possibilita a comparação do desempenho de todos os sistemas em relação à sua aplicação real [OGATA].

Visto que, para a obtenção de resultados foi, unicamente, aplicada a entrada em degrau unitário, apenas será abordada a componente teórica relativa a este tema.

3.1.1. RESPOSTA TRANSITÓRIA E RESPOSTA ESTACIONÁRIA

A resposta temporal de um sistema de controlo é constituída por duas partes, a resposta transitória e a resposta estacionária ou em regime permanente. A resposta transitória entende-se como a resposta desde o tempo inicial, $t = 0$, até o sistema atingir o valor próximo do desejado, que normalmente consiste em $\pm 2\%$ desse valor. A resposta estacionária é referente ao comportamento à saída do sistema à medida que t tende para infinito, e após o sistema estabilizar.

Desta forma, podemos descrever a resposta, $c(t)$, de um sistema como:

$$c(t) = c_{tr}(t) + c_{ss}(t)$$

onde $C_{tr}(t)$, representa a resposta transitória e $C_{ss}(t)$ a resposta em regime permanente. [OGATA].

3.1.2. RESPOSTA AO DEGRAU

O principal problema de controlo que todos os programadores enfrentam é a necessidade em manter constante a variável de saída. Considerando o exemplo de controlo de posicionamento de uma antena, conforme a Figura 27, a entrada do sistema é um degrau com amplitude igual ao valor desejado para a sua saída [TROFINO].

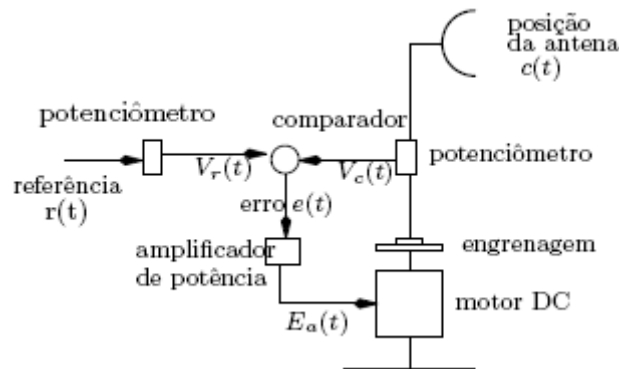


Figura 18 - Sistema de Controlo de uma Antena [TROFINO]

Quando o utilizador pretende alterar a posição da antena do seu estado inicial, posição zero, para uma nova posição, posição 1, o sinal de entrada deve ser um degrau unitário. Quando é aplicado o degrau unitário no sistema de controlo, a posição da antena vai evoluir da posição zero para a posição 1 segundo uma curva que depende de como o sistema de controlo foi projectado. Na Figura 19 são ilustradas respostas típicas dos sistemas de controlo quando se aplica um degrau unitário na sua entrada, isto para diferentes sistemas e diferentes parâmetros de controlo [TROFINO]. As resposta estão identificadas à sua esquerda através das letras a, b, c e d, e pode-se observar uma melhoria acentuada comparando a primeira curva (a) com a última (d).

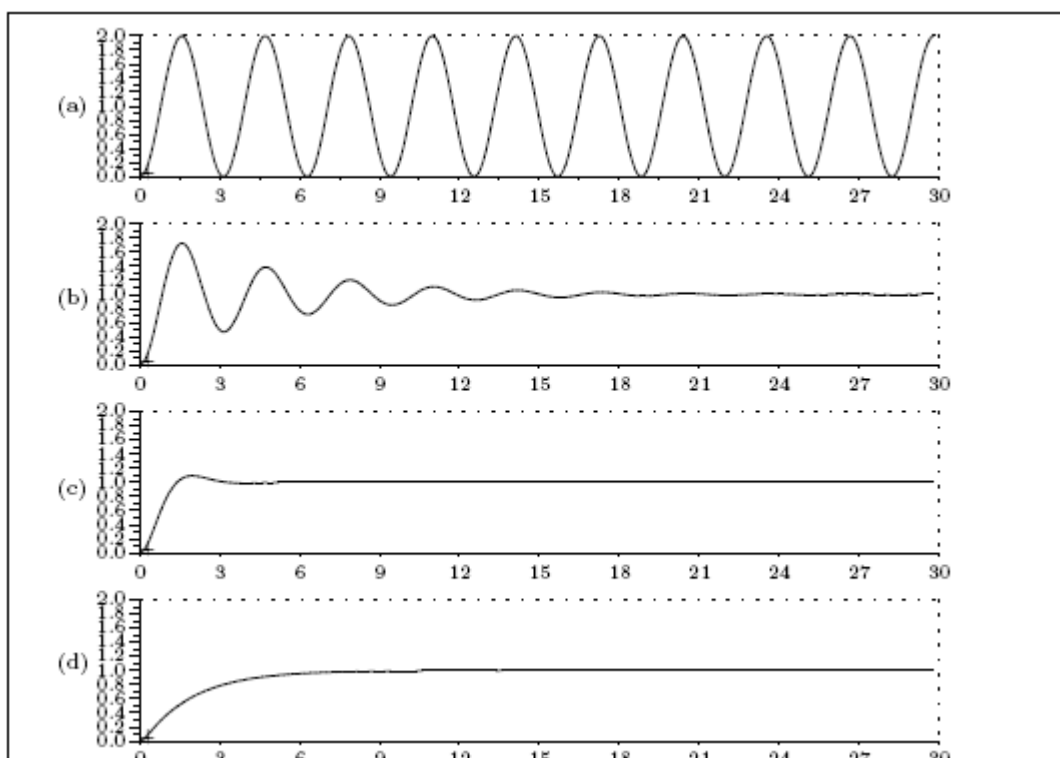


Figura 19 - Curvas típicas da resposta ao degrau unitário [TROFINO]

Normalmente deseja-se uma resposta transitória rápida, com poucas oscilações e que a variável controlada, que neste caso trata-se da posição da antena, atinja o *set point* conforme as curvas (c) ou (d), na Figura 28.

3.1.3. CARACTERIZAÇÃO DAS RESPOSTA TRANSITÓRIA

Na maior parte dos casos práticos, as características de desempenho de um sistema de controlo são classificadas em termos temporais [OGATA]. Estas especificações são bastante úteis, pois permitem a avaliação da resposta de saída no que diz respeito aos tempos que caracterizam determinado sistema. Na Figura 20 pode-se observar a curva característica de um sistema em resposta ao degrau unitário, onde estão assinalados alguns dos parâmetros mais importantes, a saber, a sobre-elevação M_p , o tempo de pico t_p , o tempo de estabelecimento t_s e o erro em regime permanente e_{ss} .

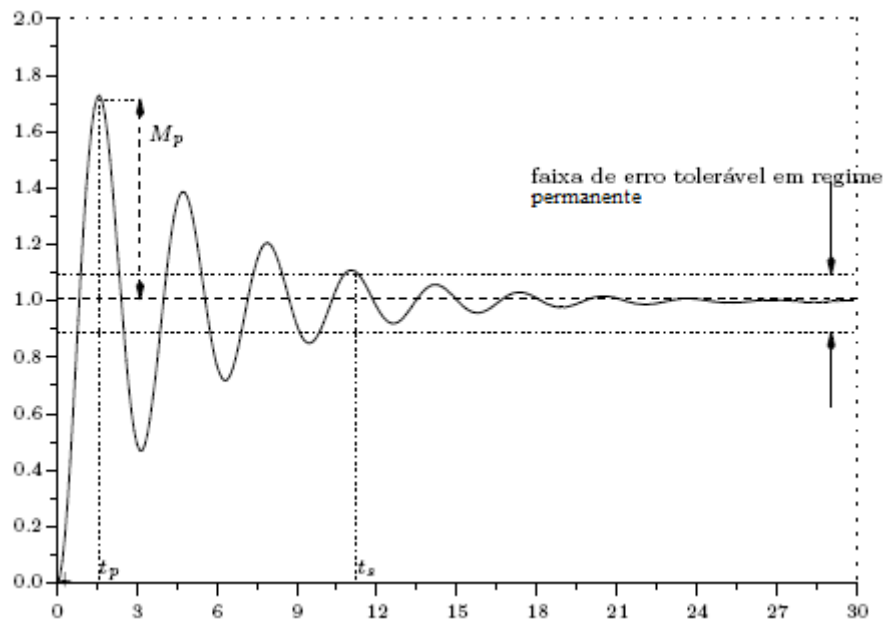


Figura 20 - Parâmetros de desempenho resultantes da entrada em degrau [TROFINO]

A resposta transitória de um sistema a uma entrada em degrau unitário depende das condições iniciais. Quando se está a comparar as respostas transitórias de diversos sistemas, normalmente é utilizada uma condição inicial padrão que obriga os sistemas em causa a estarem em repouso, ou seja, o valor da saída e a sua derivada em ordem ao tempo devem ser iguais a zero. Desta forma, é possível comparar as características das respostas transitórias dos vários sistemas [OGATA].

Na prática, antes de atingir o regime permanente, a resposta transitória de um sistema de controlo apresenta frequentemente oscilações amortecidas, o que permite retirar as informações para caracterizar o sistema em causa. Os parâmetros que normalmente servem de comparação são os seguintes [OGATA]:

1. Tempo de pico, t_p – tempo que o sistema demora até atingir o primeiro pico de sobre-elevação [TROFINO];
2. Sobre-elevação Máxima, M_p – valor relativo da diferença entre o valor máximo da saída (ao longo do tempo) e o valor da saída em regime estacionário [TROFINO]:

$$M_p = \frac{y(t_p) - y(\infty)}{y(\infty)}$$

3. Tempo de Estabelecimento, t_s – tempo necessário para que a curva da resposta do sistema alcance valores numa gama que ronda entre 2 a 5% do valor final, permanecendo nesse intervalo indefinidamente [TROFINO].
4. Tempo de Atraso, t_d – Tempo que o sistema demora para reagir ao sinal de entrada [OGATA];
5. Tempo de Subida, t_r – tempo necessário para que a resposta passe de 10% a 90%, ou de 5% a 95%, ou de 0% a 100% do valor final [OGATA].

De uma forma geral não é possível determinar as expressões matemáticas para os parâmetros de desempenho da resposta ao degrau anteriormente mencionado. No entanto, para sistemas de segunda ordem sub-amortecidos ($1 > \xi > 0$) isto é possível. Essas expressões são apresentadas a seguir.

1. Tempo de pico, t_p – o tempo de pico pode ser caracterizado como o primeiro instante de tempo em que a derivada da saída em relação ao tempo é zero. Logo t_p pode ser definido por [TROFINO]:

$$t_p = \frac{\pi}{w_d} = \frac{\pi}{w_n \sqrt{1 - \xi^2}}$$

onde w_d , w_n e ξ representam respectivamente frequência amortecida, frequência natural não amortecida e o coeficiente de amortecimento.

2. Sobre-elevação (*overshoot*) Máxima, M_p – num sistema de segunda ordem sub-amortecido, o valor da resposta ao degrau unitário em regime permanente é 1, $y(\infty) = 1$ [TROFINO].

$$M_p = \frac{y(t_p) - y(\infty)}{y(\infty)} = y(t_p) - 1 = e^{\frac{\sigma}{w_d}\pi} = e^{-\frac{\pi\xi}{\sqrt{1-\xi^2}}}$$

3. Tempo de Estabelecimento, t_s - Ao contrário do cálculo da sobre-elevação e do tempo de pico, não existe uma expressão analítica exacta para o tempo de estabelecimento, mas sim métodos de cálculo que permitem obter uma aproximação do valor de t_s .

Sendo a resposta ao degrau do sistema:

$$y(t) = 1 - \frac{e^{\sigma t}}{\sqrt{1-\xi^2}} \text{sen}(w_d t + \phi), \phi = \tan^{-1} \frac{\sqrt{1-\xi^2}}{\xi}$$

Obrigando a que amplitude do seno se situe dentro da faixa de tolerância que caracteriza o tempo de estabelecimento temos uma condição suficiente para garantir que o tempo de estabelecimento foi atingido com a respectiva tolerância. Note-se que em regime permanente o valor da saída é $y(\infty) = 1$ e a amplitude do seno tende para zero quando $t \rightarrow \infty$. Seja δ a tolerância do erro que define o tempo de estabelecimento e fazendo com que a amplitude da função seno se situe dentro da faixa de tolerância, tem-se [TROFINO]:

$$\left| \frac{y(t_s) - y(\infty)}{y(\infty)} \right| = \left| \frac{e^{\sigma t_s}}{\sqrt{1-\xi^2}} \text{sen}(w_d + \phi) \right| \leq \delta \Rightarrow t_s = \frac{\ln(\delta \sqrt{1-\xi^2})}{\sigma}$$

onde $\sigma = -\xi w_n$ é a parte real dos pólos.

Outra aproximação, mais simples, para se obter o valor do tempo de estabelecimento é fazer a analogia com um sistema de primeira ordem. Num sistema de primeira ordem o valor em regime permanente da resposta é atingido após 4 períodos de tempo com 2% de erro e após 3 períodos de tempo com 5% de erro. Para um sistema de segunda ordem podemos aproximar t_s definindo como constante de tempo $T = -1/\sigma$ e assim temos [TROFINO]:

$$t_s = 4T \text{ para } 2\% \text{ de erro; } t_s = 3T \text{ para } 5\% \text{ de erro}$$

3.1.4. PÓLOS, ZEROS E A SUA INFLUÊNCIA NA RESPOSTA

Da mesma forma que a função de transferência representa um sistema de equações diferenciais, os seus pólos e zeros definem, efectivamente, o comportamento do sistema, pelo que os seus pólos caracterizam directamente os parâmetros de uma resposta homogénea.

A resposta natural de um sistema linear SISO (*single input and single output*) tem como condições iniciais [ANALYSIS]:

$$y_h(t) = \sum_{i=1}^n C_i e^{\lambda_i t}$$

onde as constantes C_i são determinadas através das condições iniciais do sistema e os expoentes λ_i são as raízes da equação característica, ou seja, sendo a equação característica dada por:

$$D(s) = s^n + a_{n-1}s^{n-1} + \dots + a_0 = 0$$

as raízes são os pólos do sistema, $\lambda_i = p_i$.

Estas afirmações levam ao estabelecimento das seguintes relações:

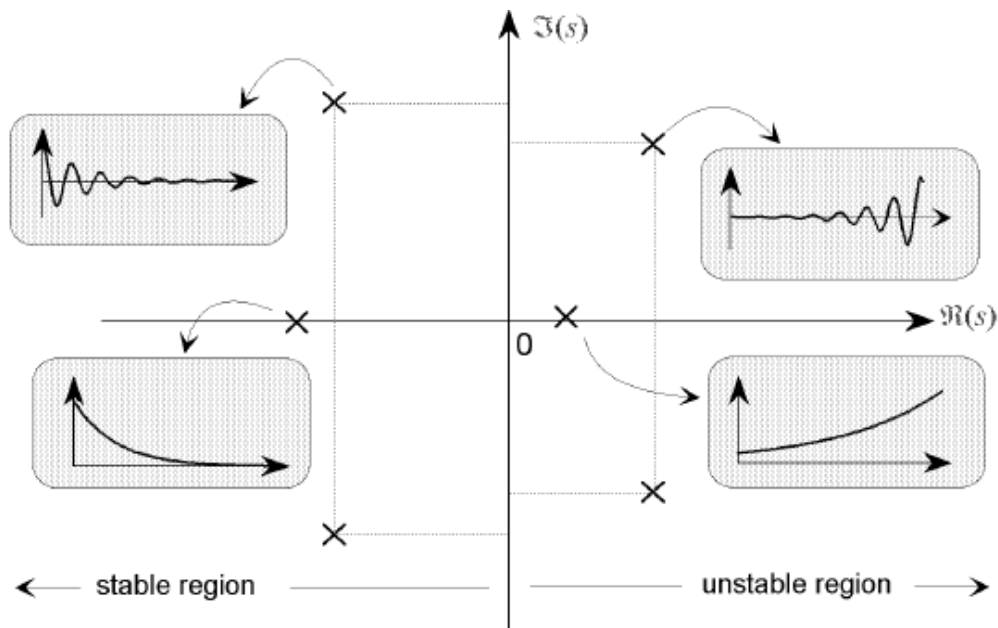


Figura 21 - Localização de pólos no plano-s [ANALYSIS]

Com a observação da localização dos pólos no plano-s obtêm-se uma série de dados sobre o comportamento do sistema em resposta à entrada em degrau.

A saída do sistema é definida pela expressão, já anteriormente apresentada:

$$y_h(t) = \sum_{i=1}^n C_i e^{p_i t}$$

Onde:

1. Um pólo real, $p_i = -\sigma$, na metade esquerda do plano-s representa um decaimento exponencial, $C e^{-\sigma t}$, na resposta do sistema. A taxa de decaimento é determinada pela localização do pólo, pelo que, quanto mais afastado da origem mais rápido

será o decaimento, pelo contrário, mais perto da origem o decaimento será mais lento.

2. Um pólo na origem $p_i = 0$ provoca uma resposta oscilatória de amplitude constante que depende das condições iniciais;
3. Um pólo real na metade direita do plano-s corresponde a um aumento exponencial da resposta, $Ce^{\sigma t}$, com tendência para a instabilidade.
4. Um pólo complexo conjugado, $\sigma \pm jw$, na metade esquerda do plano-s corresponde a uma resposta em decaimento sinusoidal, na forma de $Ae^{-\sigma t} \sin(wt + \phi)$, onde A e ϕ são determinados pelas condições iniciais. A taxa de decaimento é especificada por σ , sendo a frequência de oscilação dada por w .
5. Um par de pólos imaginários, $\pm jw$, gera uma resposta oscilatória, de amplitude constante, e determinada pelas condições iniciais.
6. Um par de pólos complexos na metade direita do plano-s gera uma resposta exponencial crescente.

3.2. SISTEMAS DE 1º ORDEM

Definição: “São sistemas de 1º Ordem, sistemas cuja função de transferência possui apenas um pólo” [TROFINO].

Consideremos o sistema de primeira ordem a seguir representado:

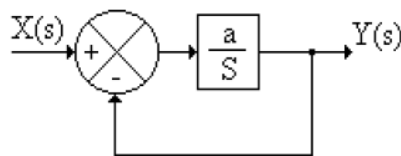


Figura 22 - Sistema de Primeira Ordem [HELIO]

cuja função de transferência é dada por:

$$\frac{Y(s)}{X(s)} = T(s) = \frac{a}{s + a}$$

O denominador da função de transferência $T(s)$ igualada a zero, é denominado como equação característica e, para que o sistema seja estável, as raízes dessa equação devem estar localizadas no semiplano esquerdo do plano-s, conforme a Figura 23.

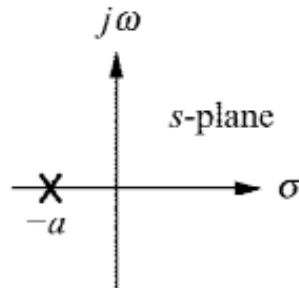


Figura 23 - Localização de um Pólo no plano-s de um Sistema de 1º Ordem [SANTOS]

Isto significa que, para o sistema exemplificado ser estável é necessário que $a > 0$, isto é, $S = -a$.

3.2.1. RESPOSTA AO DEGRAU UNITÁRIO DE UM SISTEMA DE PRIMEIRA ORDEM

Seja o sistema de primeira ordem sem zeros representado pelo seguinte bloco:

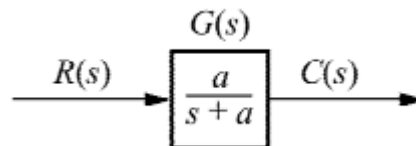


Figura 24 - Sistema de Primeira Ordem [SANTOS]

Se a entrada for um degrau unitário, onde $R(s) = \frac{1}{s}$, a transformada de Laplace da resposta ao degrau é $C(s)$, onde:

$$C(s) = R(s)G(s) = \frac{a}{s(s + a)}$$

Aplicando-se a transformada inversa, a resposta ao degrau pode ser expressa como:

$$c(t) = c_f(t) + c_n(t) = 1 - e^{-at}$$

estando representada graficamente na Figura 25.

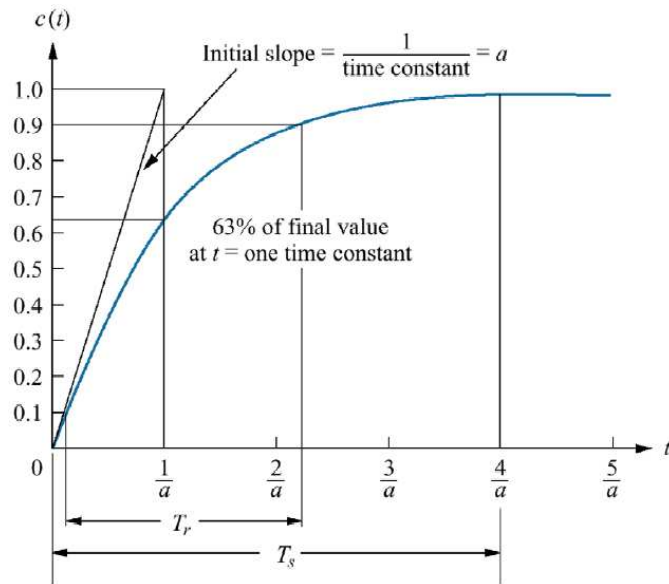


Figura 25 - Curva de Resposta à entrada em Degrau Unitário [SANTOS]

Onde o pólo de entrada na origem gerou a resposta forçada $c_f(t) = 1$, e o pólo do sistema em $-a$, conforme ilustrado na Figura 23, gerou a resposta natural $c_n(t) = e^{-at}$.

3.3. SISTEMAS DE 2º ORDEM

Definição: “Sistemas de segunda ordem são aqueles cuja função de transferência possui dois pólos” [TROFINO].

Para a sua análise será considerado o sistema ilustrado na Figura.26.

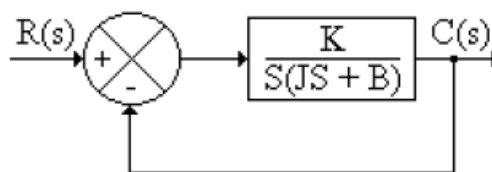


Figura 26 - Sistema de Segunda Ordem [HELIO]

sendo a sua função de transferência dada por:

$$\frac{C(s)}{R(s)} = \frac{K}{JS^2 + BS + K}$$

e onde os pólos da função são representados por:

$$S_{1,2} = -\frac{B}{2J} \pm \sqrt{\left(\frac{B}{2J}\right)^2 - \frac{K}{J}}$$

Os pólos da função de transferência mostrada acima, podem ser:

- a) Reais $\rightarrow \left(\frac{B}{2J}\right)^2 - \frac{K}{J} > 0$ (2 raízes reais)
- b) Complexos $\rightarrow \left(\frac{B}{2J}\right)^2 - \frac{K}{J} < 0$ (2 raízes complexas conjugadas)

Para a análise da resposta transitória de um sistema de segunda ordem, deve-se representar a função de transferência na sua forma canônica.

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

É de salientar que o desempenho de sistemas de segunda ordem depende somente de dois parâmetros, os quais são: a frequência natural de oscilação “ ω_n ” e o coeficiente de amortecimento “ ξ ”.

Os pólos desta função são determinados da seguinte forma:

$$S_{1,2} = -\xi\omega_n \pm j\omega_n\sqrt{1 - \xi^2}$$

Como já foi mencionado anteriormente o comportamento dinâmico de um sistema de segunda ordem pode ser descrito em termos de dois parâmetros ξ e ω_n .

3.3.1. SISTEMA SUB-AMORTECIDO $\rightarrow 0 < \xi < 1$

Se $0 < \xi < 1$, os pólos do sistema em malha fechada são complexos conjugados e situam-se no semiplano esquerdo do plano-s, pelo que o sistema é sub-amortecido, sendo a sua resposta transitória oscilatória.

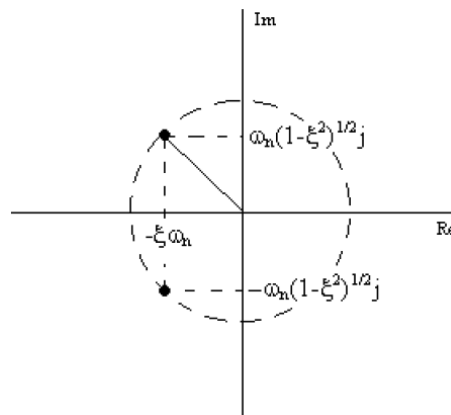


Figura 27 - Localização do Pólos num Sistema Subamortecido [EI]

Para este tipo de sistemas, a função de transferência é representada da seguinte forma:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{(s + \xi\omega_n + j\omega_d)(s + \xi\omega_n - j\omega_d)}$$

onde:

$$\omega_d = \omega_n \sqrt{1 - \xi^2}$$

A resposta do sistema pode ser visualizada na figura a seguir.

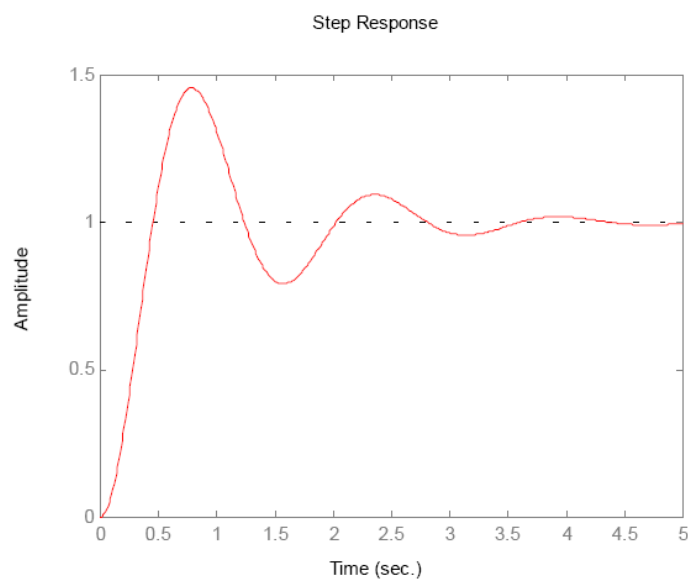


Figura 28 – Resposta de um Sistema Sub-amortecido à entrada em Degrau [EI]

3.3.2. SISTEMA NÃO-AMORTECIDO $\rightarrow \xi = 0$

Se $\xi = 0$, a resposta transitória não decai. Esta função possui um pólo na origem, proveniente do degrau unitário na entrada e, dois pólos imaginários decorrentes do sistema. Estes pólos, estão localizados sobre o eixo imaginário (Figura 29), e são responsáveis pela resposta sinusoidal cuja frequência é igual à localização do eixo imaginário.

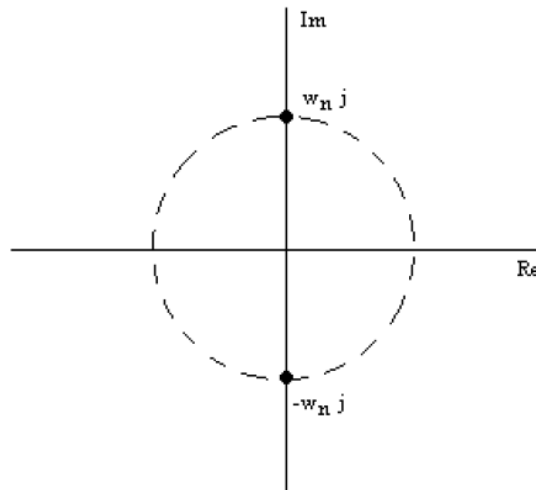


Figura 29 - Localização dos Pólos de um Sistema não amortecido [EI]

Neste tipo de sistemas a resposta é caracterizada por uma oscilação e uma amplitude constantes. Este comportamento provém da ausência de uma parte real no par de pólos o que provoca uma resposta em forma de exponencial sem decaimento. Na Figura 30 está ilustrado um exemplo de uma resposta de um destes sistemas.

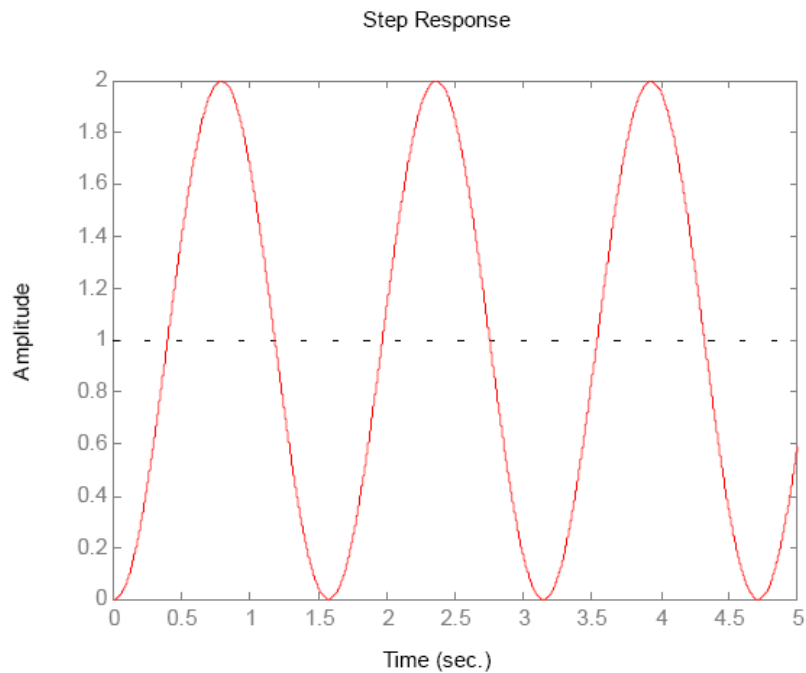


Figura 30 - Resposta de um sistema de segunda ordem à entrada em degrau não amortecido [EI]

3.3.3. SISTEMA CRITICAMENTE AMORTECIDO $\rightarrow \xi = 1$

Esta função possui um pólo na origem, proveniente do degrau unitário na entrada, e dois pólos iguais sob o eixo real (Figura 40).

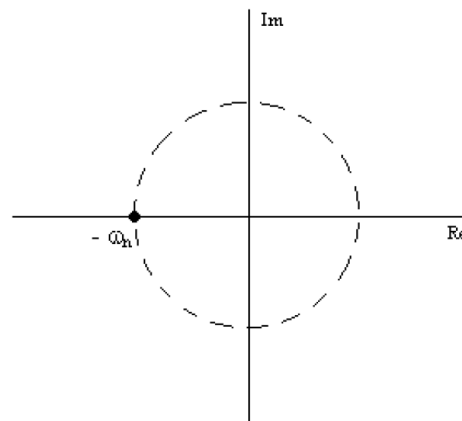


Figura 31 - Localização dos Pólos de um Sistema onde o coeficiente de amortecimento é unitário [EI]

Para este caso, a função de transferência é dada por:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{(s + \omega_n)^2}$$

Os pólos reais neste caso geram uma resposta natural que consiste numa exponencial simples e numa exponencial multiplicada pelo tempo, onde a frequência das exponenciais é igual à coordenada de localização dos pólos reais [SANTOS].

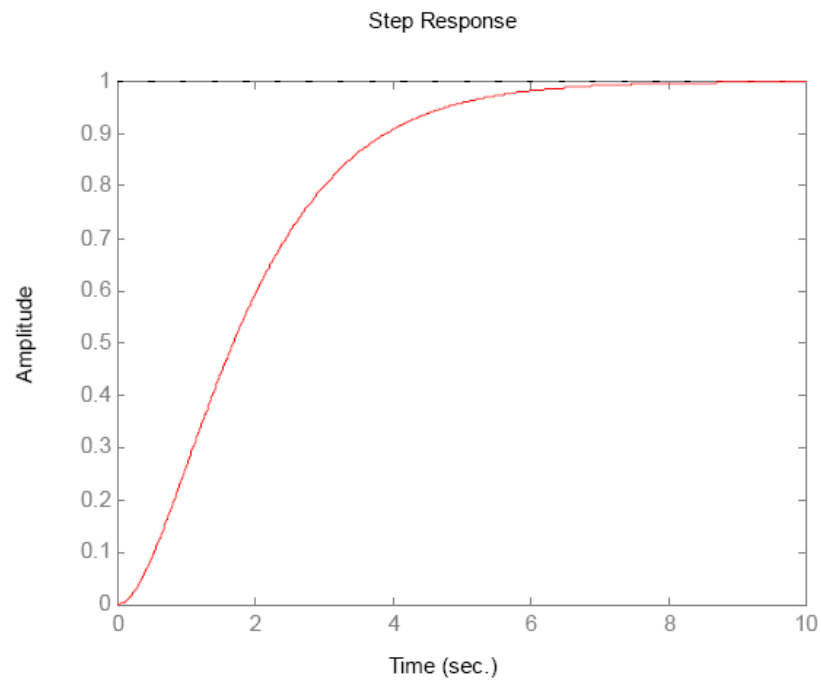


Figura 32 - Resposta de um sistema de segunda ordem à entrada em degrau criticamente amortecido [EI]

3.3.4. SISTEMA SOBRE-AMORTECIDO $\rightarrow \xi > 1$

Neste caso, o sistema apresenta dois pólos reais, negativos e diferentes. Logo, as raízes da equação característica são:

$$s_{1,2} = -\xi\omega_n \pm \omega_n\sqrt{\xi^2 - 1}$$

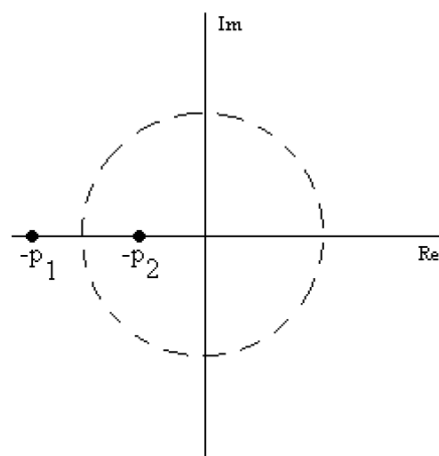


Figura 33 - Localização dos Pólos de um Sistema sobre-amortecido [EI]

A função de transferência do sistema é dada por:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{(s + \xi\omega_n + \omega_n\sqrt{\xi^2 - 1})(s + \xi\omega_n - \omega_n\sqrt{\xi^2 - 1})}$$

Sendo a curva da sua resposta ilustrada na figura a seguir:

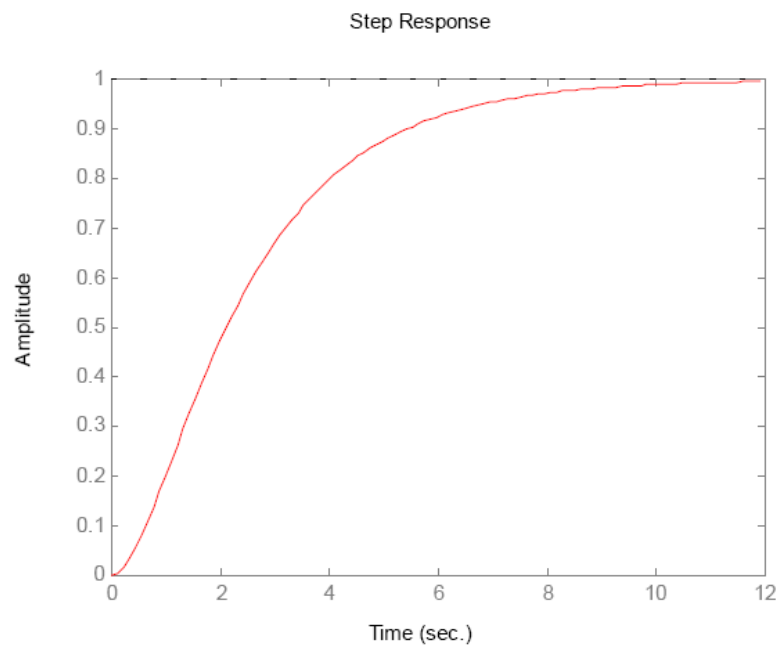


Figura 34 - Resposta de um sistema de segunda ordem à entrada em degrau com sobre-amortecimento [EI]

É importante referir que quanto maior for o valor do coeficiente de amortecimento, e para o mesmo valor de ω_n , temos $|S_1| \gg |S_2|$ e portanto o efeito do pólo S_1 sobre a resposta é anulado ao que prevalece o efeito do pólo S_2 que está localizado mais próximo do eixo imaginário. Sendo assim, para valores de $\xi \gg 1$ o sistema torna-se extremamente lento [TROFINO].

3.3.5. SISTEMA INSTÁVEL $\rightarrow \xi < 1$

Para valores negativos de ξ um dos pólos da função de transferência é positivo e portanto a saída do sistema diverge exponencialmente [TROFINO].

O gráfico a seguir ilustrado, apresenta uma família de curvas com vários valores de ξ . É de notar que as curvas que chegam mais rapidamente ao valor final, correspondem a sistemas sub-amortecidos com $0,5 < \xi < 0,8$ [HELIO].

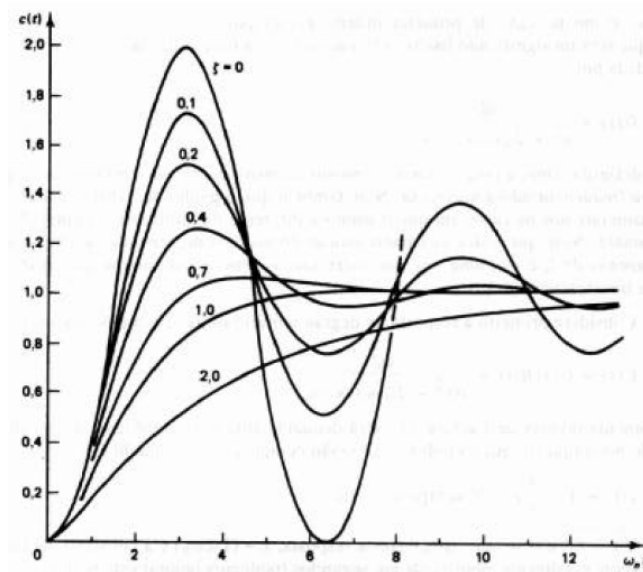


Figura 35 – Variação da resposta de um Sistema de Segunda Ordem com a variação do factor de amortecimento [EI]

3.4. SISTEMAS DE ORDEM SUPERIOR

Seja um sistema genérico representado pelo seguinte diagrama de blocos.

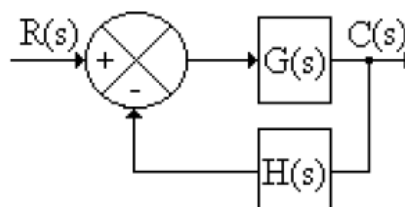


Figura 36 - Sistema de Ordem Superior [HELIO]

A função de transferência deste sistema pode ser escrita como o quociente entre dois polinômios, onde o grau do polinômio do numerador deve ser menor que o grau do polinômio do denominador.

$$\frac{C(s)}{R(s)} = \frac{b_0 s^m + b_1 s^{m-1} + \dots + b_{m-1} s + b_m}{a_0 s^n + a_1 s^{n-1} + \dots + a_{m-1} s + a_m} \quad m \leq n$$

A função de transferência acima apresentada pode ser expandida em frações parciais, como uma série de termos de primeira e de segunda ordens. Aplicando-se técnicas de factorização de polinômios à expressão anterior temos [HELIO]:

$$\frac{C(s)}{R(s)} = \frac{K(S + Z_1)(S + Z_2) \dots (S + Z_m)}{(S + P_1)(S + P_2) \dots (S + P_n)}$$

Onde Z_m representa os zeros e P_n os pólos do sistema.

Considerando que o sinal de entrada $R(s)$ é um degrau unitário, a aplicação da expansão em frações parciais na expressão acima, resulta:

$$C(s) = \frac{a}{s} + \sum_{i=1}^n \frac{a_i}{s + P_i}$$

Desta forma, uma função de transferência de ordem n é representada pelo somatório de termos de 1ª ordem e de 2ª ordem, onde cada um apresenta uma certa influência na resposta global do sistema, que depende da constante de tempo e do ganho associado ao pólo. Os termos que apresentam ganhos muito pequenos, praticamente não exercem influência na resposta transitória pelo que, podem ser desprezados. Desta forma, é possível analisar a resposta de um sistema de ordem superior a partir de um sistema simplificado. Os pólos que estão mais próximos do eixo $j\omega$ no plano- s , chamados pólos dominantes, correspondem aos termos de resposta transitória que decaem mais lentamente.

Conclui-se então que a estabilidade relativa e a resposta transitória de um sistema estão diretamente ligados com a localização de pólos e zeros no plano- s . Muitas vezes, é necessário ajustar um ou mais parâmetros do sistema para que o mesmo tenha um desempenho satisfatório [HELIO].

3.5. O EFEITO DOS ZEROS NA RESPOSTA DO SISTEMA

O efeito dos zeros sobre a resposta do sistema é mais difícil de ser definido. Apesar da localização dos zeros não influenciar a estabilidade do sistema que é determinada apenas pelos pólos, a localização dos zeros provoca alterações na resposta do sistema, pelo que podem fazer com que os resultados sejam bastante diferentes quando comparados com sistemas apenas com pólos. De facto, a introdução de um zero na resposta do sistema, pode introduzir sobreelongação caso os zeros estejam colocados no semiplano esquerdo, ou subelongação, no caso de estarem no semiplano direito.

À semelhança do que sucede com os pólos, também se pode definir zeros rápidos e zeros lentos, sendo que os rápidos são aqueles que estão bastante afastados em relação ao eixo imaginário quando comparado com os pólos dominantes. Os zeros lentos são aqueles que estão mais próximos do eixo imaginário do que os pólos dominantes. Desta localização ocorrem as sobreelongações com maior amplitude se o sistema for mais rápido e vice-versa.

Para ilustrar a influência dos zeros na resposta do sistema à entrada ao degrau unitário, utilizando pólos iguais mas com zeros diferentes, vamos considerar as funções de transferência da Tabela 3. Os sistemas definidos pelas funções de transferência $G_1(s)$, $G_2(s)$, $G_3(s)$ e $G_4(s)$ podem ser expandidos em fracções simples por [MARUYAMA]:

$$Y(s) = \frac{K_1}{(s+1)} + \frac{K_2}{(s+1+j)} + \frac{K_3}{(s+1-j)} + \frac{K_4}{s}$$

nos quais os seus parâmetros estão indicados na Tabela 3.

Tabela 3 - Tabela de funções de transferência com zeros diferentes [MARUYAMA]

	K_1	K_2	K_3	K_4
$G_1(s) = \frac{2}{(s+1)(s+1+j)(s+1-j)}$	-2	$0.5 + j0.5$	$0.5 - j0.5$	1
$G_2(s) = \frac{0.2(s+10)}{(s+1)(s+1+j)(s+1-j)}$	2	$-1.5 - j0.5$	$-1.5 - j0.5$	1
$G_3(s) = \frac{-0.2(s+10)}{(s+1)(s+1+j)(s+1-j)}$	-6	$2.5 + j0.5$	$2.5 - j0.5$	1
$G_4(s) = \frac{10(s^2+0.1s+0.2)}{(s+1)(s+1+j)(s+1-j)}$	1	$5 + j4.5$	$5 - j4.5$	1

Na figura que se segue são apresentadas as quatro respostas para os diferentes sistemas.

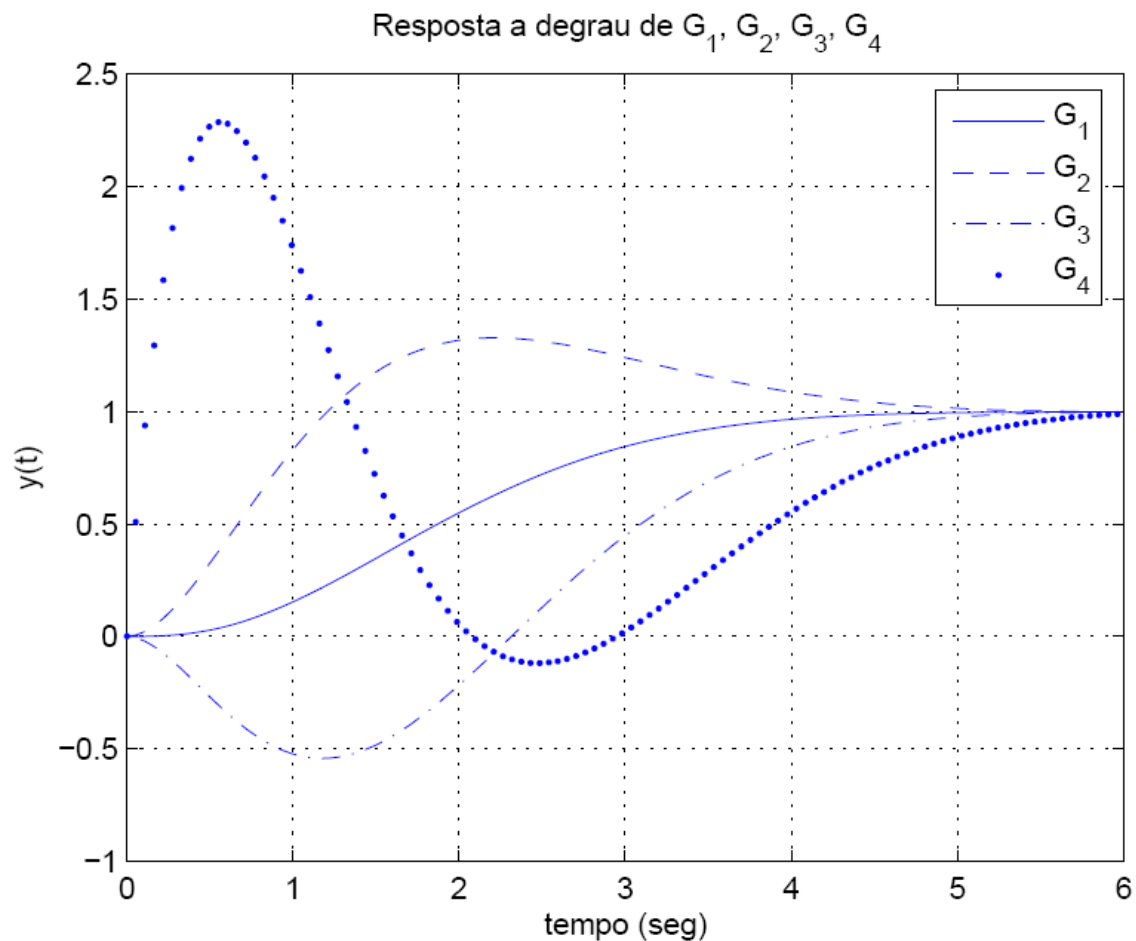


Figura 37 - Resposta de diversos sistemas à entrada em degrau variando os zeros da função transferência [MARUYAMA]

3.6. CONCLUSÕES

Resumidamente conclui-se que a base do comportamento dos sistemas está directamente ligado ao número de pólos e zeros que constituem a função de transferência e à sua respectiva localização no plano-s.

Outro ponto que caracteriza o comportamento de cada sistema é a forma como este é excitado à entrada, podendo assumir diversas respostas consoante do tipo de entrada.

Estes dois conceitos são fundamentais para se iniciar o controlo de sistemas.

4. CONTROLADOR PID

O controlador PID (Proporcional, Integral e Derivativo) é de longe o controlador com mais aplicações conhecidas. Trata-se de um controlador simples e fácil de sintonizar, que resolve de uma forma rápida e eficaz a maior parte dos problemas de controlo.

É mais utilizado em aplicações de controlo industrial, tais como na indústria da pasta do papel, controlo de processos químicos, aplicações agrícolas entre muitas outras.

Neste capítulo será descrito o controlador PID, as suas principais aplicações, as suas características e os diversos métodos de sintonia.

4.1. NOTA HISTÓRICA SOBRE O PID

No ano de 1935, *Ralph Clarridge da Taylor Instruments Companies*, desenvolveu o controlador de três termos que antecipava a variação do sinal de erro para resolver o problema de oscilação de uma malha de controlo de temperatura numa indústria de celulose. Chamaram-lhe inicialmente *per-cat*. A acção derivativa foi testada apenas em casos especiais até ao ano 1939, quando surgiu uma versão totalmente reformulada do controlador PID *fulscope* que foi proposto como padrão de sistemas de controlo na empresa. No mesmo ano, a *Foxboro Instrument Company* lança o controlador pneumático

Stabilog, que possuía a tecnologia *hyper-reset*, baseada na derivada do sinal do erro [PEREIRA].

O controlador PID demonstrou a sua importância em algumas aplicações consideradas difíceis, mas continuava com dificuldades de afirmação no mercado. Existia ainda uma grande dificuldade de difusão nos processos industriais. Os três motivos apontados eram:

- A não contemplação da controlabilidade no projecto das unidades indústrias;
- A complexidade e fragilidade dos elementos actuadores na época;
- A inexistência de regras simples para a sintonia dos parâmetros do controlador PID.

Destes motivos o mais importante é o último porque continua a ter impacto no uso do controlador, ao contrário dos dois primeiros problemas que foram ultrapassados e não se tratavam de problemas do controlador. No ano de 1942 surge a primeira tentativa bem sucedida para a resolução deste problema, com o artigo “*Optimum Settings for Automatic Controller*”, da autoria de J. G. Ziegler e N. B. Nichols, ambos da *Taylor Instrument Companies*. Neste artigo, foram apresentados dois procedimentos utilizando regras simples de sintonia dos parâmetros dos controladores, baseadas nas características dinâmicas do processo. Este artigo marca um ponto de evolução para o controlador PID. A partir do momento da divulgação destas regras de sintonia, começa a propagação do seu uso, que leva outros investigadores a interessarem-se pelo assunto e a desenvolverem novos métodos de ajuste a partir das ideias originais [PEREIRA].

4.2. INTRODUÇÃO AO CONTROLO PID

PID é o acrónimo para proporcional, integral e derivativo. Um controlador PID é definido por um controlador que aplica estas três funções [ARAKI M]. Normalmente, a sintonia de um controlador PID é feita com o próprio inserido na malha de controlo, o que originou uma série de métodos e regras de sintonia. Estes métodos e regras são uma vantagem enorme em relação aos outros controladores pois permitem ajustes finos de controlo em tempo real, sem a necessidade de parar o processo ou reiniciá-lo. Para além disso, tem vindo a ser desenvolvidos métodos de sintonia automática que permitem a sintonia automática *online* do controlador [OGATA].

De uma forma simples, o primeiro passo para a introdução de um controlador PID no sistema será a escolha de qual o tipo de controlador a ser aplicado. Poderá conter apenas a componente proporcional (P), a proporcional e a integral (PI), a proporcional e a derivativa

(PD) ou a conjugação dos parâmetros proporcional, integral e derivativa (PID). Uma vez escolhido o tipo de controlador, o passo seguinte será ajustar os vários parâmetros do controlador, o que consiste na dedução de determinados valores de resposta, quando este é sujeito a entradas específicas. Na Figura 38 está ilustrada a configuração de um controlador PID em malha fechada [LOURENÇO].

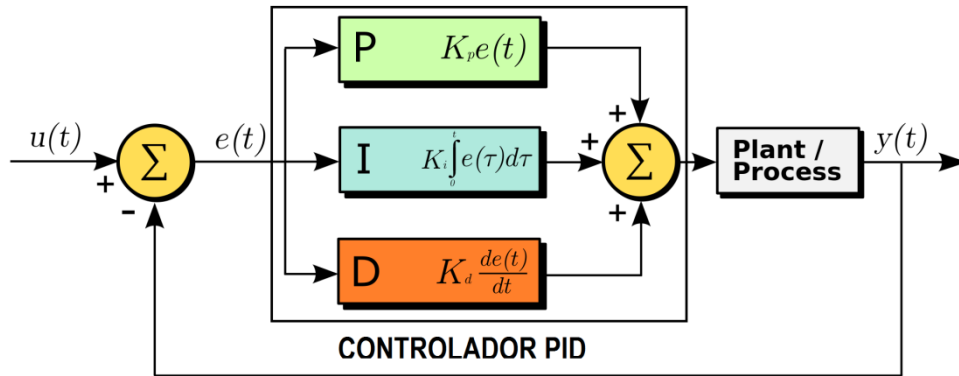


Figura 38 - Diagrama de blocos do Controlador PID [http://en.wikipedia.org/wiki/File:PID_en.svg]

O objectivo do controlador consiste em manter a saída do processo $y(t)$ no valor desejado ou de referência $u(t)$ (*set-point*), eliminando continuamente o erro, $e(t)$. Para isso o controlador irá aplicar continuamente a acção de controlo à entrada do processo [PEREIRA].

Esta acção é composta pela soma dos termos proporcional, integral e derivativo do erro, onde o erro é a diferença entre o sinal de entrada e o sinal de saída, $e(t) = u(t) - y(t)$, sendo as variáveis K_p , K_I e K_D os parâmetros do controlador, nomeadamente o ganho proporcional, ganho integrativo e o ganho derivativo, respectivamente [PEREIRA].

Assim sendo temos:

$$\text{Acção de Controlo} = K_p * e(t) + K_I * \int_0^t e(t)dt + K_D * \frac{de(t)}{dt}$$

Este tipo de controlador não necessita saber o modelo matemático do sistema a controlar, o qual na maior parte das vezes pode ser bastante difícil de determinar, por isso, de um ponto de vista económico, quando o custo de sintonia for inferior ao custo associado à análise do

sistema e projecto de um controlador, é preferível recorrer aos métodos de sintonia do PID [LOURENÇO].

4.2.1. ACÇÃO PROPORCIONAL

Aplicando apenas a acção proporcional obtemos o seguinte diagrama:

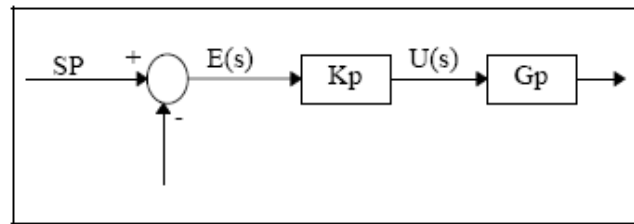


Figura 39 - Diagrama de Blocos de um Controlador P [LOURENÇO]

A expressão que o caracteriza é a seguinte:

$$u(t) = K_p e(t) \xrightarrow{L} U(s) = K_p E(s)$$

em que K_p é designado por ganho proporcional.

Basicamente, este tipo de controlador não é mais do que um amplificador com ganho ajustável. O aumento deste parâmetro pode minimizar o erro em regime permanente (e_{ss}), contudo, também pode, por consequência, aumentar o tempo de estabelecimento e eventualmente levar o sistema à instabilidade. De uma forma geral, pode-se afirmar que este tipo de controlador só se aplica quando o ganho proporcional é suficientemente elevado para reduzir o erro em regime estacionário (e_{ss}) a um nível aceitável [LOURENÇO].

4.2.2. ACÇÃO PROPORCIONAL-ÍNTegral

Para um controlador PI temos o seguinte diagrama de blocos:

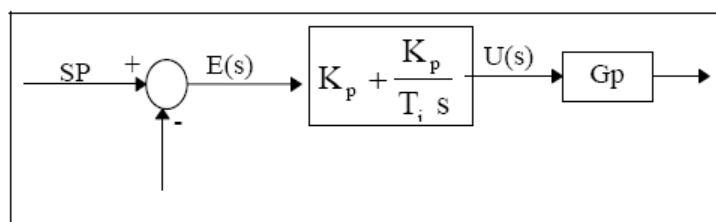


Figura 40 - Diagrama de Blocos de um Controlador PI [LOURENÇO]

A expressão que o caracteriza é a seguinte:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau \right] \xrightarrow{L} U(s) = K_p \left(1 + \frac{1}{T_i s} \right) E(s)$$

em que T_i é designado por tempo integral e representa o tempo necessário para que a contribuição da acção integral iguale a acção proporcional.

As curvas que representam este tipo de controlador são apresentadas na Figura 41.

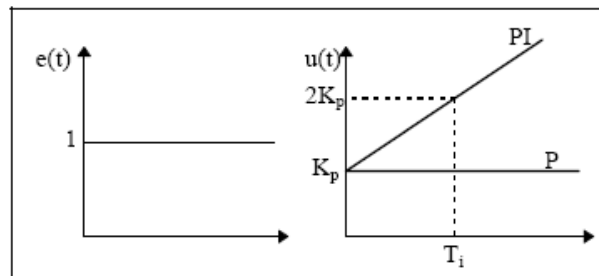


Figura 41 - Curva da resposta de saída para um controlador PI [LOURENÇO]

O efeito prático após introdução do parâmetro integral no controlador é a eliminação do erro em regime permanente (e_{ss}), independentemente do sistema que se pretende controlar. Matematicamente, a eliminação e_{ss} consegue-se através da adição de um pólo na origem da função de transferência do controlador.

Se, por um lado, a acção integral elimina o erro em regime permanente (e_{ss}), por outro, aumenta o tempo de estabelecimento e piora a estabilidade relativa. Para equilibrar a resposta, o ganho proporcional deverá ser reduzido.

Normalmente o controlador PI é aplicado em sistemas com frequentes alterações dos seus parâmetros, quando o controlador P não é suficiente para reduzir o erro em regime estacionário a um nível aceitável. [LOURENÇO]

4.2.3. ACÇÃO PROPORCIONAL-DERIVATIVA

A representação do diagrama de blocos de um controlador PD é ilustrada na Figura 42.

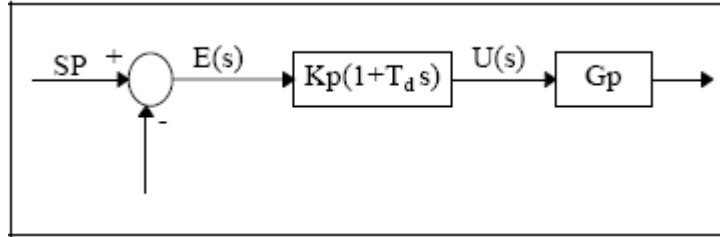


Figura 42 -Diagrama de blocos de um controlador PD [LOURENÇO]

A expressão que caracteriza este controlador é a seguinte:

$$u(t) = K_p \left[e(t) + T_D \frac{de(t)}{dt} \right] \xrightarrow{L} U(s) = K_p (1 + T_D s)$$

em que T_D é o tempo derivativo e representa a antecipação da acção derivativa relativamente à acção proporcional.

As curvas da resposta deste tipo de controlador estão ilustradas na Figura 43.

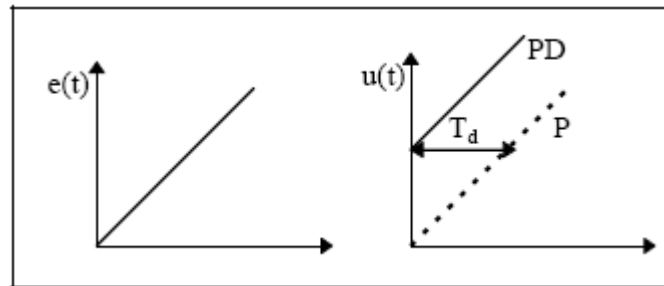


Figura 43 - Curva da resposta de saída para um controlador PD [LOURENÇO]

Neste caso, a acção de controlo irá ser proporcional à taxa de variação do erro, o que implica que o modo derivativo não possa ser usado sozinho. Este controlador será muito mais sensível do que os anteriores controladores (P e PI), pois como responde à variação do erro, permite correcções antes de este ser elevado.

Não obstante, o facto de o modo derivativo não afectar directamente o e_{ss} , adiciona amortecimento ao sistema, melhorando a estabilidade e, desta forma permite a utilização de valores de K_p mais elevados e, diminuindo o e_{ss} [LOURENÇO].

4.2.4. ACÇÃO PROPORCIONAL-INTEGRA-DERIVATIVA

O controlador com as três acções de controlo é denominado controlador PID e, é representado pelo seguinte diagrama de blocos:

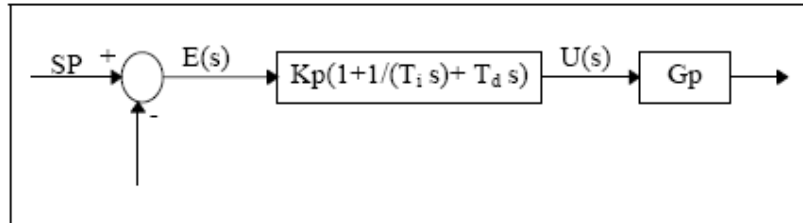


Figura 44 - Diagrama de blocos de um controlador PID [LOURENÇO]

A expressão que caracteriza o controlador PID é:

$$u(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right] \xrightarrow{L} U(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) E(s)$$

Num controlador PID o parâmetro integral é utilizado para eliminar o erro em regime permanente (e_{ss}) proveniente de grandes variações dos parâmetros do sistema. O parâmetro derivativo, com o seu efeito estabilizador, provoca o aumento do ganho e a redução das oscilações, o que leva a uma resposta mais rápida quando comparado com o controlador P e PI [LOURENÇO].

Para facilitar a sintonia de um sistema apenas com base no efeito que cada parâmetro produz na resposta, pode-se resumir esses mesmos efeitos na seguinte tabela.

Tabela 4 - Resumo dos efeitos de cada parâmetro P, I e D [LOURENÇO]

Resposta	Tempo de Subida	Sobre-elevação	Tempo de Estabelecimento	Erro em Regime Permanente
Proporcional	Diminuição	Aumento	Sem alteração	Diminuição
Integral	Diminuição	Aumento	Aumento	Elimina
Derivativo	Sem alteração	Diminuição	Diminuição	Sem alteração

Contudo, esta tabela só deve ser usada como referência, pois uma alteração num parâmetro produz efeitos nos restantes [LOURENÇO].

4.3. ESCOLHA DO TIPO DE CONTROLADOR

A escolha do controlador ideal depende fortemente das características do sistema a controlar e das condições em que o controlador se insere, pois estes dois factores em conjunto influenciarão os parâmetros da resposta, tais como, o e_{ss} máximo, a sobre-elevação máxima e o tempo de estabelecimento máximo aceitáveis. Se o e_{ss} não é permitido, então o modo integral deverá ser incluído no controlador, pois é o único parâmetro capaz de eliminá-lo ou reduzi-lo. Se a resposta do sistema é lenta e apresenta alguma sobre-elevação a componente derivativa deverá ser introduzida. Caso o e_{ss} seja mínimo e não influencie a operação de todo o processo, então é possível omitir o parâmetro integral, e o uso do modo derivativo depende entre outros factores da necessidade ou não de adicionar ganho suplementar ao modo proporcional.

Pode-se então resumir que, ao adicionar-se o modo proporcional obtêm-se determinado tempo subida, se adicionarmos o modo derivativo obtêm-se uma determinada sobre-elevação e que o modo integral só deve introduzido para eliminar o e_{ss} .

Após estar definido o tipo de controlador, torna-se agora necessário recorrer a métodos empíricos para calcular os parâmetros do controlador, tais como o de Ziegler e Nichols, onde fórmulas matemáticas permitem uma aproximação bastante satisfatória quando se pretende estabilizar um processo.

Conclui-se então, que não é possível afirmar com toda a certeza que um controlador PID é melhor do que um PI ou um PD, pois isto depende da aplicação em causa. Mas pode-se afirmar que, o melhor controlador é o que for mais simples e que satisfaça a resposta desejada [LOURENÇO].

4.4. MÉTODO PRÁTICOS DE SINTONIA DE CONTROLADORES PID

4.4.1. MÉTODO DE ZIEGLER-NICHOLS (ZNM)

Em 1942, Ziegler e Nichols propuseram dois métodos clássicos para a sintonia de controladores PID, baseados na resposta experimental a uma excitação de entrada ou no valor de K_p que resulta numa estabilidade limite quando se utiliza unicamente a acção de controlo proporcional. Com base nas observações das respostas transitórias de um determinado processo a controlar, Ziegler-Nichols, propuseram um conjunto de regras para calcular o valor do ganho proporcional K_p , da constante de tempo integral T_i e da constante de tempo derivativa T_d . Em ambos os métodos de sintonia pretende-se obter uma sobre-elevação máxima de 25% na resposta à entrada em degrau [YASSER].

4.4.1.1. PRIMEIRO MÉTODO DE ZIEGLER-NICHOLS (MÉTODO DA CURVA REACÇÃO)

No primeiro método de sintonia é introduzida uma entrada em degrau no processo em malha aberta e, caso este não possua integradores nem pólos dominantes complexos conjugados, a curva de resposta ao degrau assemelha-se a uma curva em S, tal como é ilustrado na Figura 45. É importante referir que caso a resposta do sistema não se assemelhe à curva em S, este método não poderá ser aplicado [YASSER].

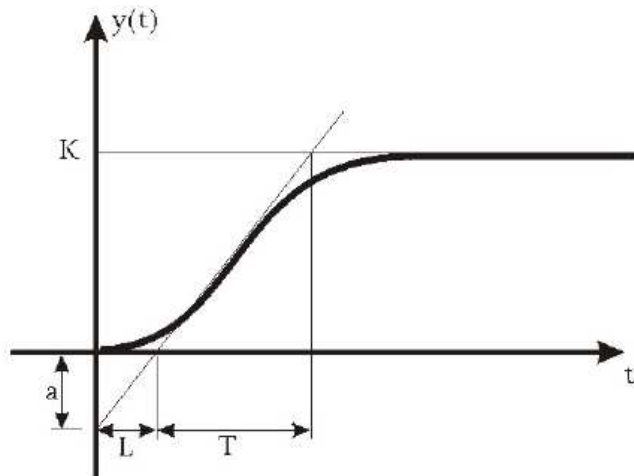


Figura 45 - Resposta ao degrau unitário para o primeiro método de Ziegler-Nichols [YASSER]

Tal como é ilustrado na Figura 45, pode-se retirar três parâmetros importantes para a determinação dos valores P, I e D. Eles são o tempo de atraso L e as constantes de tempo a e T . O tempo de atraso e as constantes de tempo são determinados traçando-se uma recta tangente à curva S no ponto de inflexão e determinando-se as intersecções com o eixo dos

tempos. Desta forma, Ziegler e Nichols conseguiram resumir as fórmulas de cálculo dos parâmetros do controlador conforme a Tabela 5.

Tabela 5 - Regras de Sintonia de Ziegler-Nichols (Primeiro Método) [YASSER]			
Tipo de Controlador	K_p	T_i	T_d
P	$\frac{1}{a}$	∞	0
PI	$0,9 \frac{1}{a}$	3L	0
PID	1,2 / a	2L	0,5L

Pode-se então apresentar a equação que representa o controlador PID através do primeiro método das regras de Ziegler-Nichols [YASSER]:

$$G_c(s) = 0,6 \frac{L \left(s + \frac{1}{L} \right)^2}{a s}$$

4.4.1.2. SEGUNDO MÉTODO DE ZIEGLER-NICHOLS (MÉTODO DA SENSIBILIDADE LIMITE)

Este método é baseado no ajuste do ganho proporcional, K_p , em malha fechada até se obterem oscilações com amplitude constante. É utilizado um conjunto de fórmulas para determinar os parâmetros do controlador, as quais requerem duas medidas do sistema: o ganho crítico (G_u : o ganho mínimo que torna o processo criticamente estável), e o período de oscilação correspondente, P_u [LOURENÇO].

O primeiro passo para colocar o sistema em oscilação constante com o ganho mínimo, é reduzir a acção integral e derivativa do controlador. Seguidamente deverá aumentar-se o ganho até que a saída do sistema entre em oscilações com amplitude constante, conforme a Figura 46.

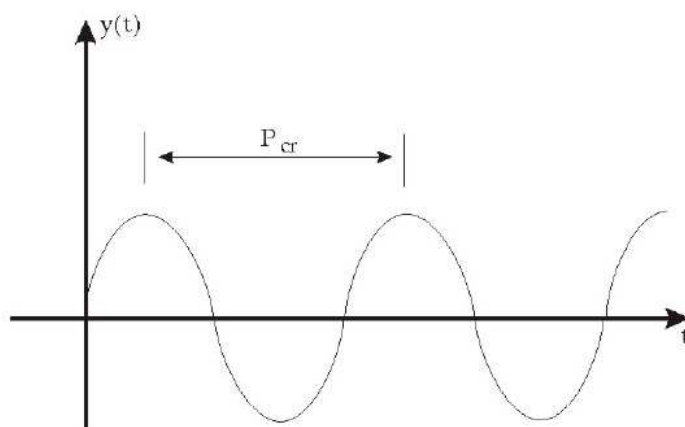


Figura 46 - Resposta oscilatória obtida através do segundo método de ZN [YASSER]

Neste ponto deve-se registrar o ganho crítico (G_u) e o período crítico (P_u) para poder-se calcular os parâmetros do controlador com base nas fórmulas apresentadas na Tabela 6 [LOURENÇO].

Tabela 6 - Regras de Sintonia de Ziegler-Nichols (Segundo Método) [LOURENÇO]

Controlador	Fórmulas		
P	$K_p = 0.5 G_u$		
PI	$K_p = 0.45 G_u$	$T_i = P_u/1.2$	
PID	$K_p = 0.6 G_u$	$T_i = P_u/2.0$	$T_d = P_u/8$

4.5. CONCLUSÕES

Conclui-se com este capítulo que o controlador PID é de simples aplicação, os seus diversos métodos de sintonia são bastante fáceis de compreender, oferecendo resultados razoáveis para a maioria dos sistemas. Podemos também concluir que o controlador PID é bastante completo quando se pretende controlar sistemas relativamente simples, de fácil modelização matemática e que o seu comportamento seja facilmente previsível, no entanto, quando o desafio aumenta este tipo de controlador apresenta algumas dificuldades em dar boas respostas.

5. DESENVOLVIMENTO DO CONTROLADOR DIFUSO

Este capítulo tem como principal objectivo dar a conhecer todo o sistema de controlo criado durante este processo de pesquisa e desenvolvimento. Será descrito em detalhe cada componente que compõe o controlador, desde a sua arquitectura até ao nível de *software*.

Ao nível da sua arquitectura será abordado cada componente que compõe o controlador lógico difuso e como cada elemento que o constitui influencia o seu desempenho. Ao nível de *software* serão apresentadas as tabelas de regras lógicas difusas que representam a lógica de controlo e ditam o seu comportamento em todas as situações.

5.1. SISTEMA DE CONTROLO

O controlador desenvolvido neste trabalho possui o melhor de dois controladores bem conhecidos, a saber, o Controlador Lógico Difuso e o Controlador PID.

Na figura que se segue, é ilustrado um diagrama de blocos simplificado do modelo do controlador criado, com vista a controlar os sistemas propostos.

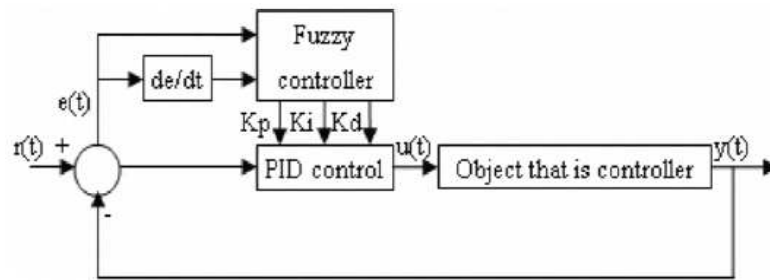


Figura 47 - Modelo do Controlador *Fuzzy*-PID [BOILER]

O sucesso deste tipo de controlador está na sintonia automática do controlador PID, pois este recebe os valores dos parâmetros K_p , K_i e K_d provenientes do controlador difuso, sendo estes valores gerados automaticamente de acordo com as regras de controlo implementadas.

Desta forma, consegue-se desenvolver um controlador com propriedades da Lógica Difusa que posteriormente usa o seu conhecimento para efectuar um controlador PID automático.

5.1.1. DIAGRAMA DE BLOCOS

O diagrama de blocos apresentado na figura abaixo representa o modelo de controlo desenvolvido com vista a controlar qualquer tipo de sistema que se introduza no bloco “*Transfer Fcn*”.

Este diagrama foi criado a partir do *Simulink* do Matlab, o qual representa um *software* de grande utilidade pois permite a simulação de processos reais bem conhecidos de uma forma simples e rápida, sem a necessidade de construir na realidade o modelo dos processos.

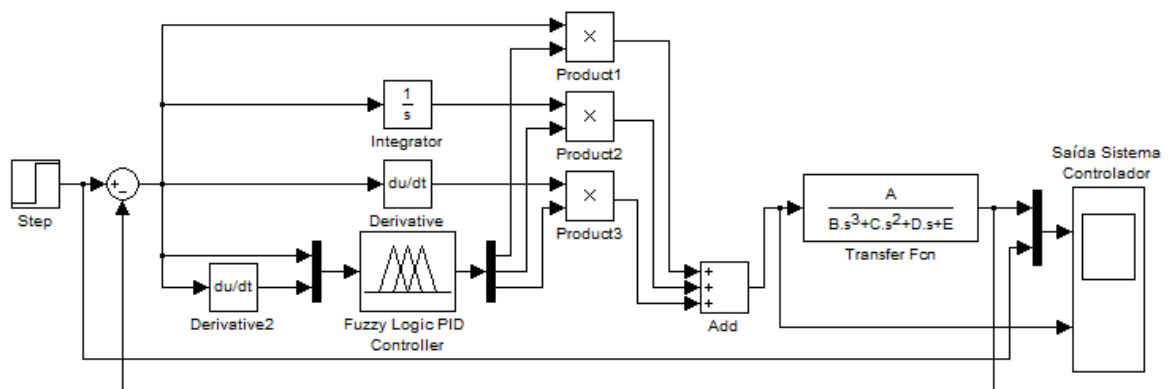


Figura 48 - Diagrama de Blocos que ilustra o Sistema de Controlo utilizado

O bloco “*Step*” representa a entrada em degrau. Este bloco poderá ser programado para “disparar” a entrada em degrau num tempo específico, onde também poderá ser definido o seu valor inicial e o seu valor final. Estas são algumas das propriedades deste bloco e podem ser observadas na Figura 49.



Figura 49 - Bloco "Step"

Os quatro tipos de blocos que serão apresentados nas linhas seguintes representam as funções matemáticas que permitem trabalhar os sinais de entrada e de saída de forma a recolher as informações desejadas para o controlo.

O primeiro bloco é denominado bloco “*Derivative*” o qual aplica a função derivada ao sinal de erro. O segundo bloco é denominado “*Integrator*”, o qual aplica a função matemática integral ao sinal de erro. O terceiro bloco, “*Product*”, realiza o produto entre dois ou mais sinais de entrada. O quarto e último bloco, chamado “*Add*”, realiza a soma entre dois ou mais sinais de entrada. Na figura seguinte são apresentadas as quatro funções atrás descritas.

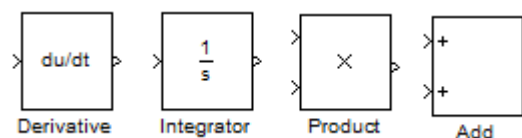


Figura 50 – Funções matemáticas representadas através de blocos

O bloco, denominado por “*Scope*” é o bloco que permite a visualização em tempo real de três tipos de sinais distintos (para este caso). São eles, a saída do sistema, a entrada em degrau e a saída do controlador difuso. A figura que se segue exemplifica o tipo de visualização que é fornecida pelo bloco “*Scope*”.

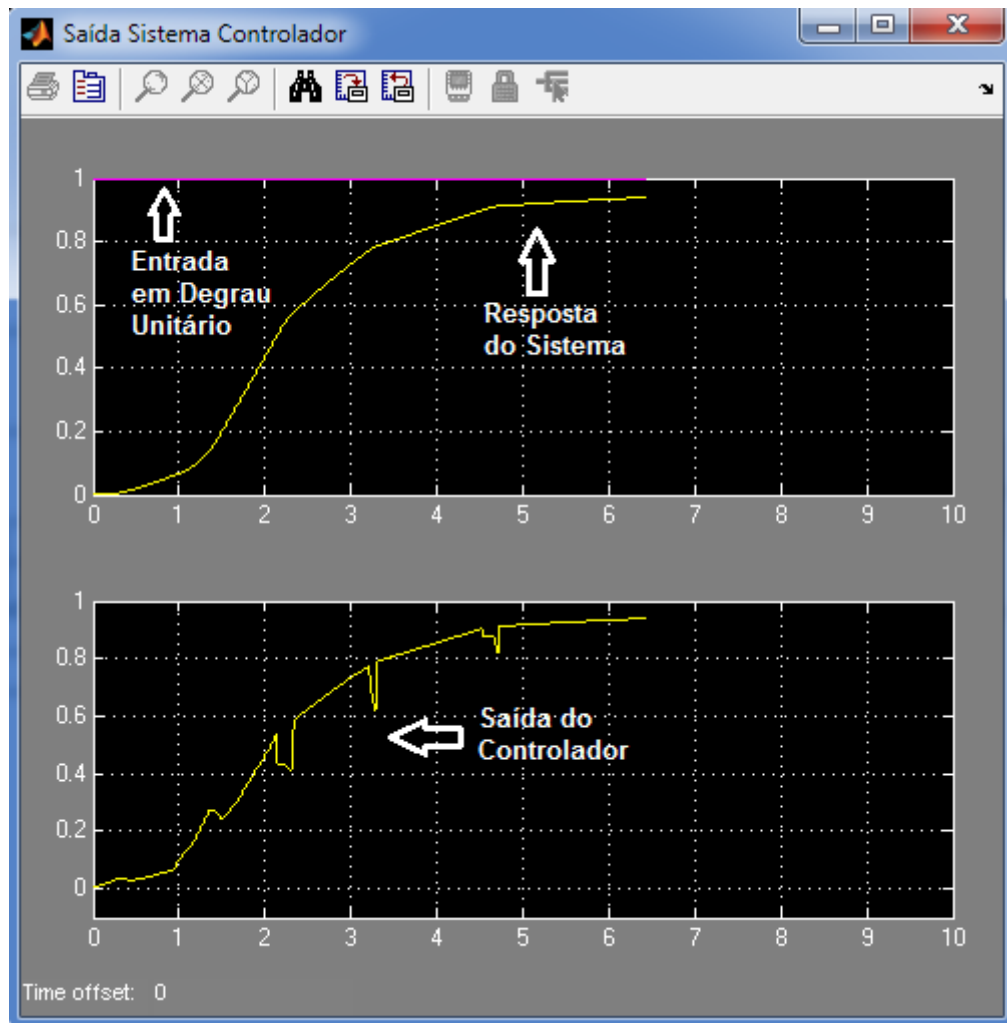


Figura 51 - Visualização da resposta de um sistema através do *Scope*

Ter acesso a este tipo de informação é vital quando se está a programar o controlador. A visualização da resposta do sistema ou da saída do controlador em determinado instante de tempo permite efetuar correções ao controlador de forma a melhorar o seu desempenho.

5.1.2. BLOCO FUZZY

O bloco *Fuzzy* representa o controlador lógico difuso. Pode-se afirmar que este bloco representa o cérebro do controlador. Nele estão implícitas as suas regras de controlo, os seus valores de controlo para as entradas e saídas, e a forma como estas vão influenciar ao longo do tempo o comportamento do sistema a controlar.

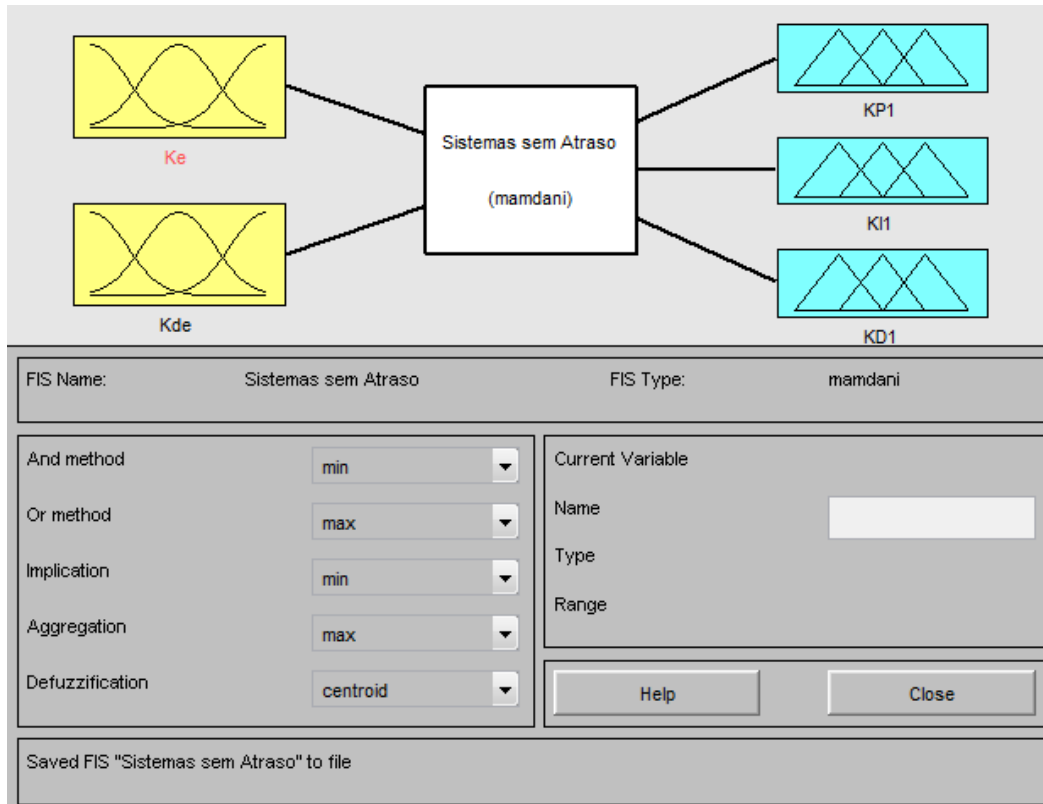


Figura 52 - Entradas e Saídas do Controlador Difuso

Conforme é mostrado na Figura 52, o controlador desenvolvido é caracterizado por duas entradas, K_e e K_{de} , e três saídas K_{P1} , K_{I1} e K_{D1} . As entradas do controlador representam respetivamente o erro e a derivada do erro ao longo do tempo. As três saídas representam os parâmetros de controlo do controlador PID, a saber, proporcional, integral e derivativo, que está localizado a jusante do bloco *fuzzy* na malha de controlo.

5.1.3. ENTRADAS E SAÍDAS DO CONTROLADOR DIFUSO

A Figura 53 representa uma interface do Matlab que permite definir manualmente a variável erro (delimitada a vermelho na figura), nomeadamente a gama de valores em que a variável erro vai situar-se, e mediante esses mesmos valores atribuir uma função de pertença (Capítulo 2). Para o controlador desenvolvido o erro está definido entre -1 e 1, sendo caracterizado por sete funções de pertença, “NB”, “NM”, “NS”, “ZE”, “PS”, “PM” e “PB”, que significam *Negative Big*, *Negative Medium*, *Negative Small*, *Zero*, *Positive Small*, *Positive Medium*, *Positive Big*, respectivamente. Estas sete funções de pertença são válidas para as cinco variáveis que constituem o controlador Fuzzy.

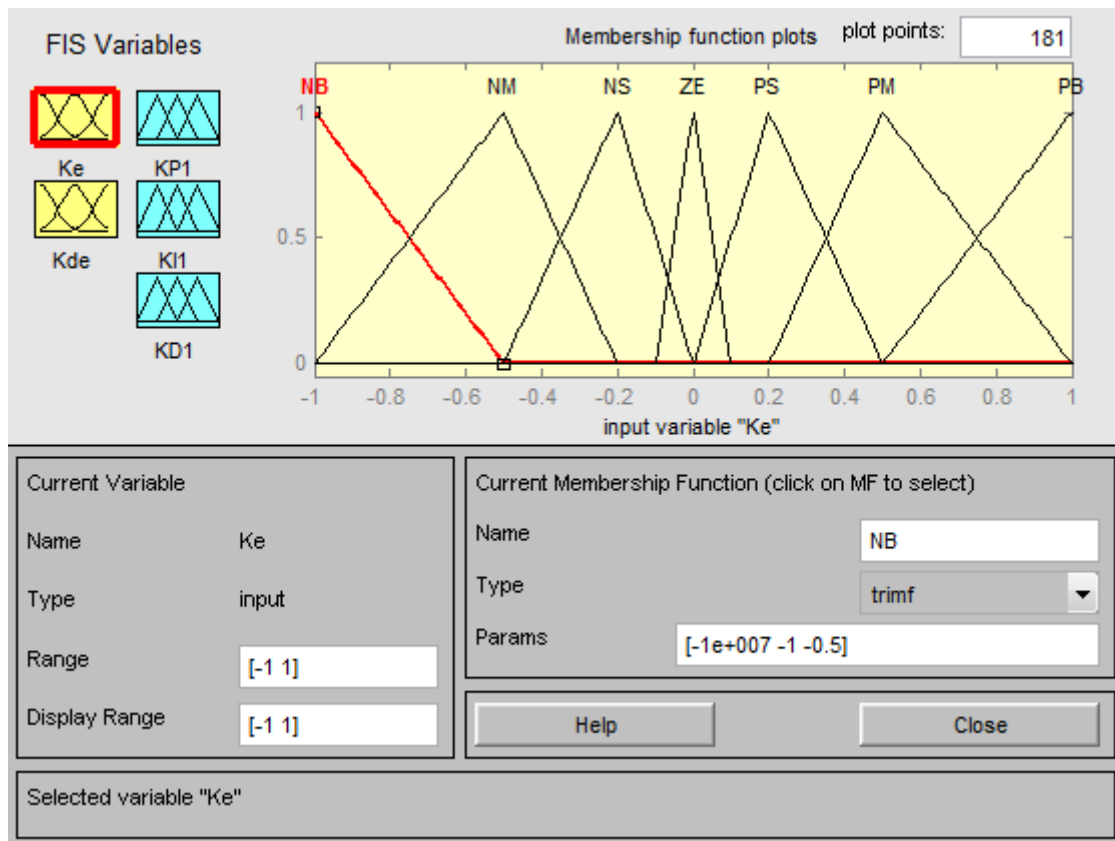


Figura 53 - Definição da variável de entrada Erro, ke

Neste trabalho, e para os conjuntos difusos correspondentes às variáveis de entrada, foram utilizadas funções de pertença triangulares (ver Figura 12).

À semelhança da variável de erro “Ke”, os valores que definem a varável “Kde” foram atribuídos de igual modo, como é observável na Figura 54.

Em termos práticos a derivada do erro não é mais do que o valor do declive em cada período de tempo. Este valor é bastante útil para o controlo de sistemas porque é interpretado pelo controlador como uma tendência, podendo ser ascendente, descendente ou estável, o que permite tomar decisões antecipadamente de forma a evitar comportamentos indesejados.

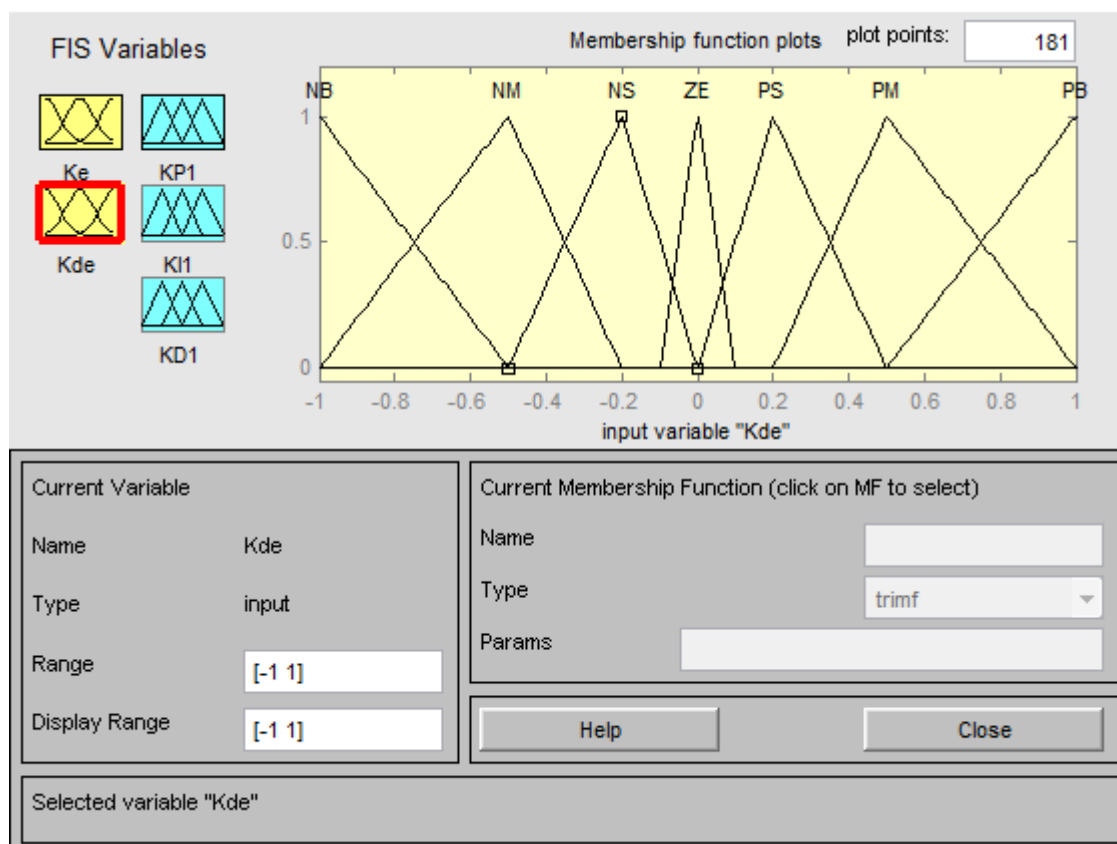


Figura 54 - Definição da variável de entrada Derivada do Erro, Kde

A escolha da gama de valores para o erro e a sua derivada não seguiu nenhum conceito teórico em particular, tendo sido escolhida com base nos resultados obtidos durante as inúmeras simulações levadas a cabo durante todo o estudo.

Uma das três variáveis de saída do bloco Fuzzy é a variável “KP1”. Esta variável representa o ganho proporcional gerado pelo controlador que, posteriormente é multiplicado pelo erro do sistema em cada período de tempo (conforme a Figura 48, secção 5.1.1). Este valor proporcional é gerado automaticamente pelo controlador Fuzzy e, assume os valores definidos pelas regras difusas num intervalo de -1 a 1, conforme é ilustrado na Figura 55.

Outra variável de saída, que neste caso representa a componente integral, é denominada por “KI1”. Esta variável assumirá valores que irão multiplicar pelo valor integral do erro, cujo resultado representa o ganho integral para o controlador.

Por último, temos a variável “KD1” que é responsável por assumir os valores derivativos, que seguindo a lógica anterior serão multiplicados pela derivada do erro.

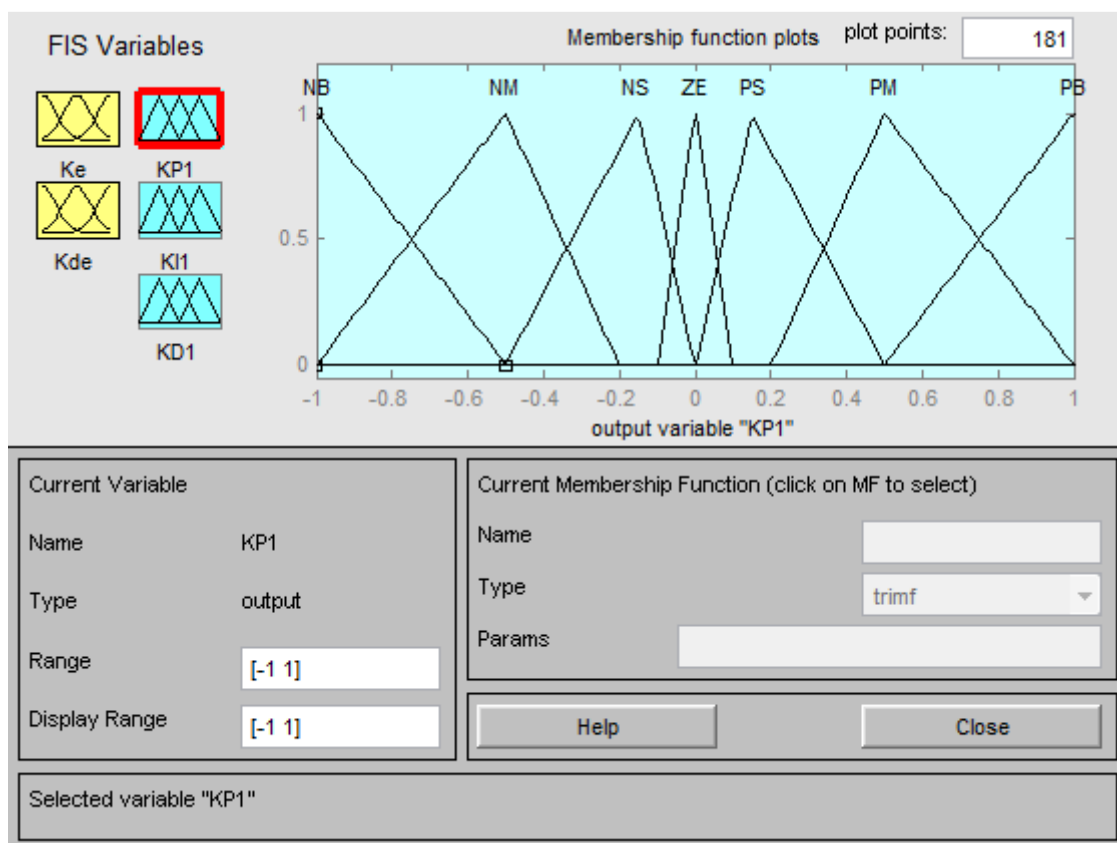


Figura 55 - Definição das variáveis de saída KP1, KI1, KD1

As gamas de valores atribuídas para as variáveis KP1, KI1 e KD1, seguiram uma lógica do tipo tentativa erro, pelo que, foram ajustadas mediante os resultados obtidos. Também aqui, a função que se utilizou para as variáveis de saída foi a triangular.

5.1.4. REGRAS DIFUSAS DE CONTROLO

De forma a interligar as variáveis de entrada, Ke e Kde, e as variáveis de saída KP1, KI1 e KD1, foram criadas as seguintes tabelas de regras que representam a lógica difusa de controlo apresentadas respectivamente nas tabelas 6, 7 e 8. De acordo com o estado instantâneo do sistema a ser controlado, as regras difusas irão em conjunto proporcionar as três componentes de controlo mencionadas anteriormente, onde cada uma tem um papel distinto no efeito que tem sobre o sistema, sendo que em conjunto, as 49 regras, quando somadas representam um sinal contínuo de controlo que controlará qualquer sistema.

Regras de Sintonia para Sistemas sem Atraso

Tabela 7 - Regras difusas para a variável de saída KP1 (sem atraso)

KP		<i>Derivada do Erro</i>						
		<i>NB</i>	<i>NM</i>	<i>NS</i>	<i>ZE</i>	<i>PS</i>	<i>PM</i>	<i>PB</i>
<i>Erro</i>	<i>NB</i>	PB	PB	PB	PB	PM	PS	PS
	<i>NM</i>	PB	PB	PB	PM	PM	PS	ZE
	<i>NS</i>	PB	PB	PB	PM	PS	ZE	ZE
	<i>ZE</i>	PB	PB	PM	PS	ZE	NS	NS
	<i>PS</i>	PM	PM	PS	ZE	ZE	NS	NS
	<i>PM</i>	PM	PS	ZE	NS	NS	NS	NM
	<i>PB</i>	PS	PS	NS	PB	NS	NM	NM

Tabela 8 - Regras difusas para a variável de saída KI1 (sem atraso)

KI		<i>Derivada do Erro</i>						
		<i>NB</i>	<i>NM</i>	<i>NS</i>	<i>ZE</i>	<i>PS</i>	<i>PM</i>	<i>PB</i>
<i>Erro</i>	<i>NB</i>	NB	NB	NM	NM	NS	ZE	ZE
	<i>NM</i>	NB	NB	NM	NS	NS	ZE	ZE
	<i>NS</i>	NB	NM	NS	NS	ZE	PS	PS
	<i>ZE</i>	NM	NM	NS	ZE	PS	PM	PM
	<i>PS</i>	NS	NS	ZE	PS	PS	PM	PB
	<i>PM</i>	ZE	ZE	PS	PS	PM	PB	PB
	<i>PB</i>	ZE	ZE	PS	PM	PM	PB	PB

Tabela 9 - Regras difusas para a variável de saída KD1 (sem atraso)

KD		Derivada do Erro						
		NB	NM	NS	ZE	PS	PM	PB
Erro	NB	PS	NS	NB	NB	NB	NM	PS
	NM	PS	NS	NB	NM	NM	NS	ZE
	NS	ZE	NS	NM	NM	NS	NS	ZE
	ZE	ZE	NS	NS	NS	NS	NS	ZE
	PS	ZE	ZE	ZE	ZE	ZE	ZE	ZE
	PM	PB	NS	PS	PS	PS	PS	PB
	PB	PB	PM	PM	PM	PS	PS	PB

Como exemplo das regras definidas apresentam-se abaixo alguns exemplos de regras com um peso mais significativo no controlo dos sistemas sem atraso.

- *Se o erro for PB e a derivada do erro for ZE então a variável de saída KP1 vale PB, KII vale PM e KD1 vale PM. (Regra a)*

Esta regra (Regra a) é a primeira regra de controlo que o sistema utiliza, pois trata-se do instante inicial em que o erro é máximo, assume-se como 1, e a derivada do erro é zero pois o sistema encontra-se parado.

- *Se o erro for ZE e a derivada do erro for ZE então a variável de saída KP1 vale PS, KII vale ZE e KD1 vale NS. (Regra b)*

A regra acima apresentada (Regra b) ilustra o estado de estabilização do sistema, em que o valor do erro e a sua derivada valem zero, respectivamente. Ao contrário da regra anterior, esta é a última a ser “chamada” pelo controlado, visto que se trata do último estado desejável para a resposta do sistema.

Imaginando que o sistema, através de alguma interferência exterior, tende a entrar numa oscilação divergente, entrando numa espiral instável foram criadas as quatro regras que se seguem (Regras c, d, e, f) foram criadas com o objectivo de controlar o sistema em situações limite.

- *Se o erro for NB e a derivada do erro for NB então a variável de saída KP1 vale PB, KII vale NB e KD1 vale PS. (Regra c)*

- Se o erro for NB e a derivada do erro for PB então a variável de saída KP1 vale PS, KI1 vale ZE e KD1 vale PS. (Regra d)
- Se o erro for PB e a derivada do erro for NB então a variável de saída KP1 vale PS, KI1 vale ZE e KD1 vale PB. (Regra e)
- Se o erro for PB e a derivada do erro for PB então a variável de saída KP1 vale NM, KI1 vale PB e KD1 vale PB. (Regra f)

Contudo, não é de todo desejável que o sistema entre nos campos limite das regras.

Regras de Sintonia para Sistemas com Atraso

Tabela 10 - Regras difusas para a variável de saída KP1 (com atraso)

KP		Derivada do Erro						
		NB	NM	NS	ZE	PS	PM	PB
Erro	NB	PB	PB	PB	PB	PM	PS	PS
	NM	PB	PB	PB	PM	PM	PS	ZE
	NS	PB	PB	PB	PM	PS	ZE	ZE
	ZE	PS	PS	PS	PS	PS	PS	PS
	PS	PS	PS	PS	PS	PS	PS	PS
	PM	PS	PS	PS	PS	PS	PS	PS
	PB	PS	PS	PS	PS	PS	PS	PS

Tabela 11 - Regras difusas para a variável de saída KI1 (com atraso)

KI		Derivada do Erro						
		NB	NM	NS	ZE	PS	PM	PB
Erro	NB	NB	NB	NM	NM	NS	ZE	ZE
	NM	NB	NB	NM	NS	NS	ZE	ZE
	NS	NB	NM	NS	NS	ZE	PS	PS
	ZE	PS	PS	PS	PS	PS	PS	PS
	PS	PS	PS	PS	PS	PS	PS	PS
	PM	PS	PS	PS	PS	PS	PS	PS
	PB	PS	PS	PS	PS	PS	PS	PS

Tabela 12 - regras difusas para a variável de saída KD1 (com atraso)

KD		Derivada do Erro						
		NB	NM	NS	ZE	PS	PM	PB
Erro	NB	PS	NS	NB	NB	NB	NM	PS
	NM	PS	NS	NB	NM	NM	NS	ZE
	NS	ZE	NS	NM	NM	NS	NS	ZE
	ZE	PS	PS	PS	PS	PS	PS	PS
	PS	PS	PS	PS	PS	PS	PS	PS
	PM	PS	PS	PS	PS	PS	PS	PS
	PB	PS	PS	PS	PS	PS	PS	PS

Para os sistemas com atraso, o princípio que se utilizou para o desenvolvimento das regras foi tendo por base a observação das respostas dos diversos sistemas, e ajustando-se cada um individualmente, sendo que no final as regras satisfazem todos os sistemas simulados.

- *Se o erro for PB e a derivada do erro for ZE então a variável de saída KP1 vale PS, KII vale PS e KD1 vale PS. (Regra g)*

À semelhança das regras de sintonia para sistemas sem atraso, esta é a primeira regra (regra g) de controlo que o sistema utiliza. Para o caso dos sistemas com atraso é importante que o valor de controlo nos instantes iniciais não seja tão elevado com a vista a que o sistema não apresente *overshoot*.

A regra que se segue (Regra h), é a regra “final”, ou a que o sistema utilizará em último quando o sistema já se encontra estabilizado no valor desejado.

- *Se o erro for ZE e a derivada do erro for ZE então a variável de saída KP1 vale PS, KII vale PS e KD1 vale PS. (Regra h)*

Para situações limite, as quatro regras que se seguem (regras i, j, k, l) foram criadas com o objectivo de controlar o sistema nessa situação. Assim, caso este, através de alguma interferência exterior, entre numa oscilação divergente.

- *Se o erro for NB e a derivada do erro for NB então a variável de saída KP1 vale PB, KII vale NB e KD1 vale PS. (Regra i)*

- Se o erro for NB e a derivada do erro for PB então a variável de saída KPI vale PS, KII vale ZE e KDI vale PS. (Regra j)
- Se o erro for PB e a derivada do erro for NB então a variável de saída KPI vale PS, KII vale PS e KDI vale PS. (Regra k)
- Se o erro for PB e a derivada do erro for PB então a variável de saída KPI vale PS, KII vale PS e KDI vale PS. (Regra l)

5.1.5. CONTROLADOR PID

De uma forma matemática a equação que se segue descreve o comportamento que se pretende para o controlador desenvolvido é dada por:

$$u(t) = KP1 * e(t) + KI1 * \int_0^t e(\tau) d\tau + KD1 * \frac{d}{dt} e(t)$$

onde " $e(t)$ ", " $\int_0^t e(\tau) d\tau$ " e " $\frac{d}{dt} e(t)$ " representa o erro, o integral do erro e a derivada do erro, sendo o erro a diferença de valores entre o *setpoint* e o valor instantâneo do sistema a controlar. A respectiva multiplicação destes parâmetros pelos parâmetros gerados pelo controlador difuso, KP1, KII e KDI, resultam nos três parâmetros de controlo desejados, proporcional, integral e derivativo, sendo que a soma dos mesmos se traduz no sinal de controlo $u(t)$.

O diagrama de blocos desenvolvido a partir do *simulink* do Matlab, que possibilita estas operações é ilustrado na figura a seguir.

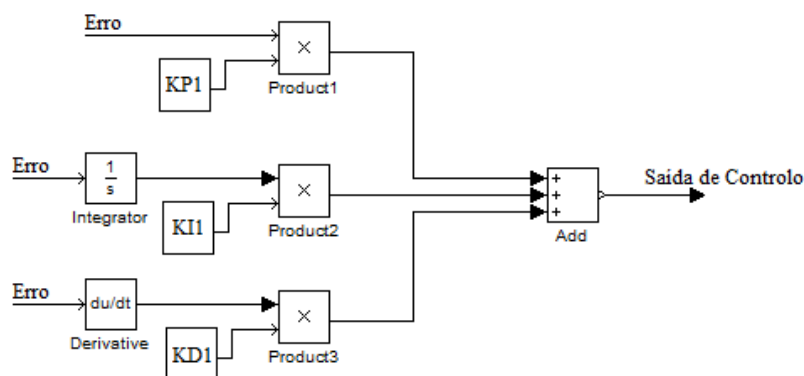


Figura 56 - Controlador PID

6. RESULTADOS OBTIDOS

Neste capítulo serão apresentados todos os resultados obtidos nas diversas simulações realizadas através do *simulink* do Matlab e que permitiram validar o algoritmo de controle proposto, para vários sistemas físicos (sistemas de 1ª e de 2ª ordem) e para funções gerais de 3ª ordem. As simulações levadas a cabo tiveram incidência sobre os sistemas de primeira, segunda e terceira ordem, com e sem atraso, simulados através do controlador FuzzyPID e para termo de comparação o controlador PID. Os resultados são apresentados sob a forma gráfica para uma análise e interpretação lógica amigável, onde estará patente a diferença entre ambos os controladores.

Para a sintonia do controlador PID foi utilizado o segundo método de sintonia de Ziegler-Nichols, denominado por método “sensibilidade limite”, tendo sido aplicado em todos os sistemas sem atraso. Para os sistemas com atraso o método utilizado foi o “tentativa-erro”, onde se foi melhorando as respostas consoante os resultados obtidos, tendo por base os parâmetros obtidos para os sistemas sem atraso.

Os resultados obtidos serão comentados tendo em consideração três de alguns pontos fundamentais na análise de sistemas, a saber: o “tempo de subida, t_r ”, “*overshoot*, M_p ” e o “tempo de estabelecimento, t_s ”.

6.1. SISTEMAS SEM ATRASO

6.1.1. SISTEMA DE CONTROLO DE NÍVEL NUM TANQUE

A função de transferência que descreve este sistema é representativa de um sistema de primeira ordem sendo dada por:

$$TF = \frac{300}{1500s + 1}$$

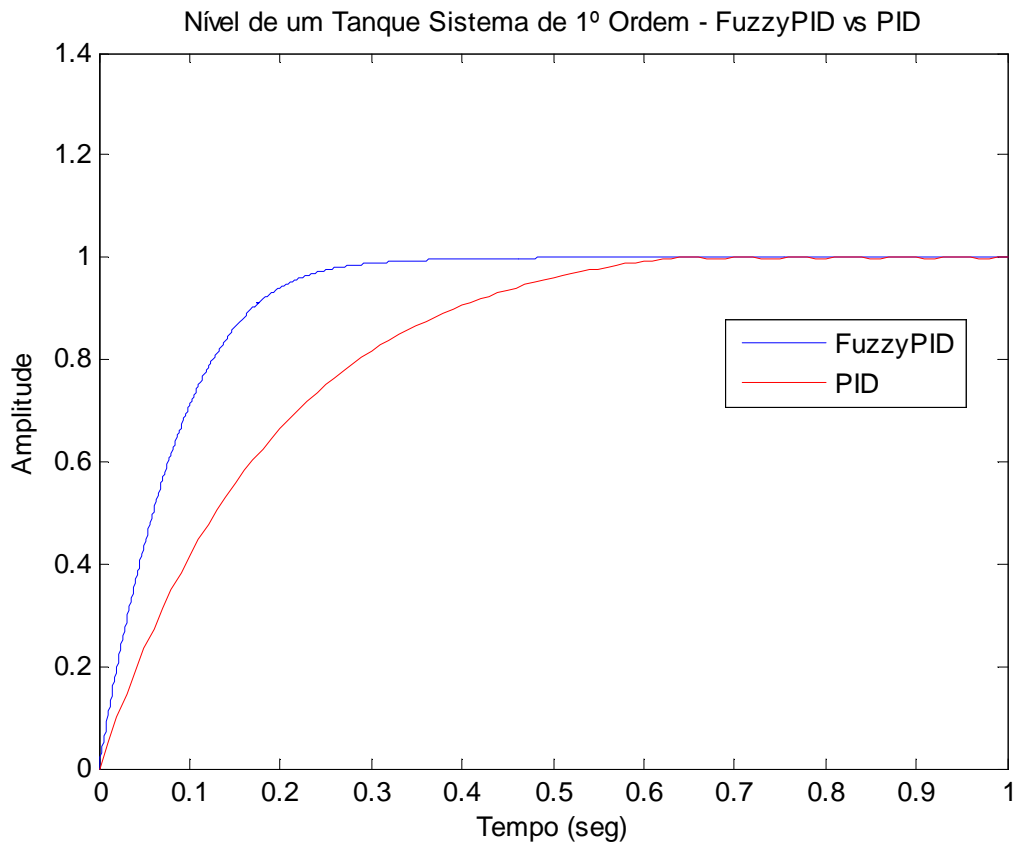


Figura 57 - Análise temporal de um sistema de controlo de nível num tanque

No sistema de controlo de nível num tanque acima apresentado, podemos observar que utilizando o controlador FuzzyPID obtêm-se uma resposta de qualidade superior à obtida com o controlador PID. Em termos de tempo de subida, o FuzzyPID necessita de menos tempo para atingir os 95% da resposta final face ao PID, sendo que nenhum dos sistemas apresentam *overshoot* e, por último, a diferença do tempo de estabelecimento entre o FuzzyPID e o PID é aproximadamente metade, como pode ser constatado na Tabela 13 apresentada na página seguinte.

Tabela 13 - Valores dos parâmetros temporais do sistema nível num tanque

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 100]	[-2 2]	[-1 1]	0,35s
PID	25	10000	0,5	0,65s

6.1.2. SISTEMA HIDRÁULICO

A função de transferência que descreve este sistema também é descrita por um sistema de 1ª ordem, sendo dada por:

$$TF = \frac{3,2}{300s + 1}$$

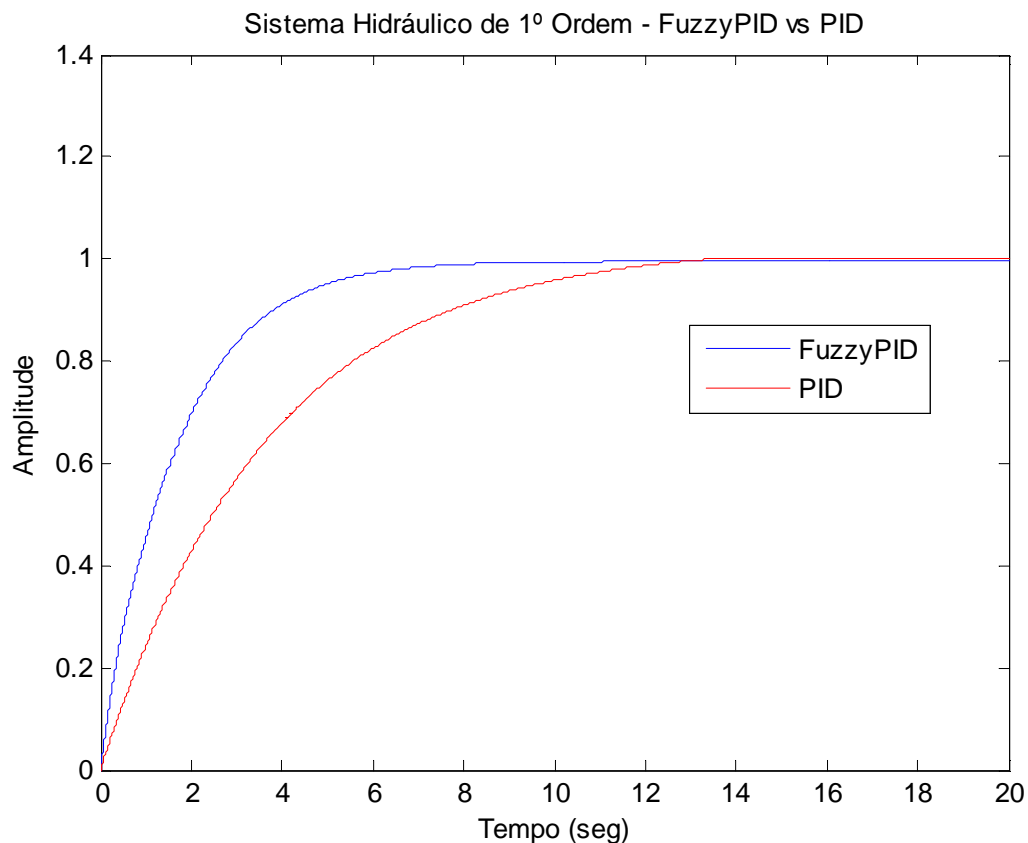


Figura 58 - Análise temporal de um sistema hidráulico

Para o sistema hidráulico acima apresentado, os resultados obtidos para os três pontos de avaliação são claramente melhores para o controlador FuzzyPID. O tempo de subida e o tempo de estabelecimento são menores para o FuzzyPID e relativamente ao *overshoot* nenhum dos controladores o apresenta.

Tabela 14 - Valores dos parâmetros temporais do sistema hidráulico

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 100]	[-2 2]	[-1 1]	8,5s
PID	25	10000	0,5	13,5s

6.1.3. SISTEMA MECÂNICO DE UMA MOLA

Este sistema é representado por uma função de transferência de 2ª ordem expressa por:

$$TF = \frac{14,9691}{s^2 + 4s + 14,9691}$$

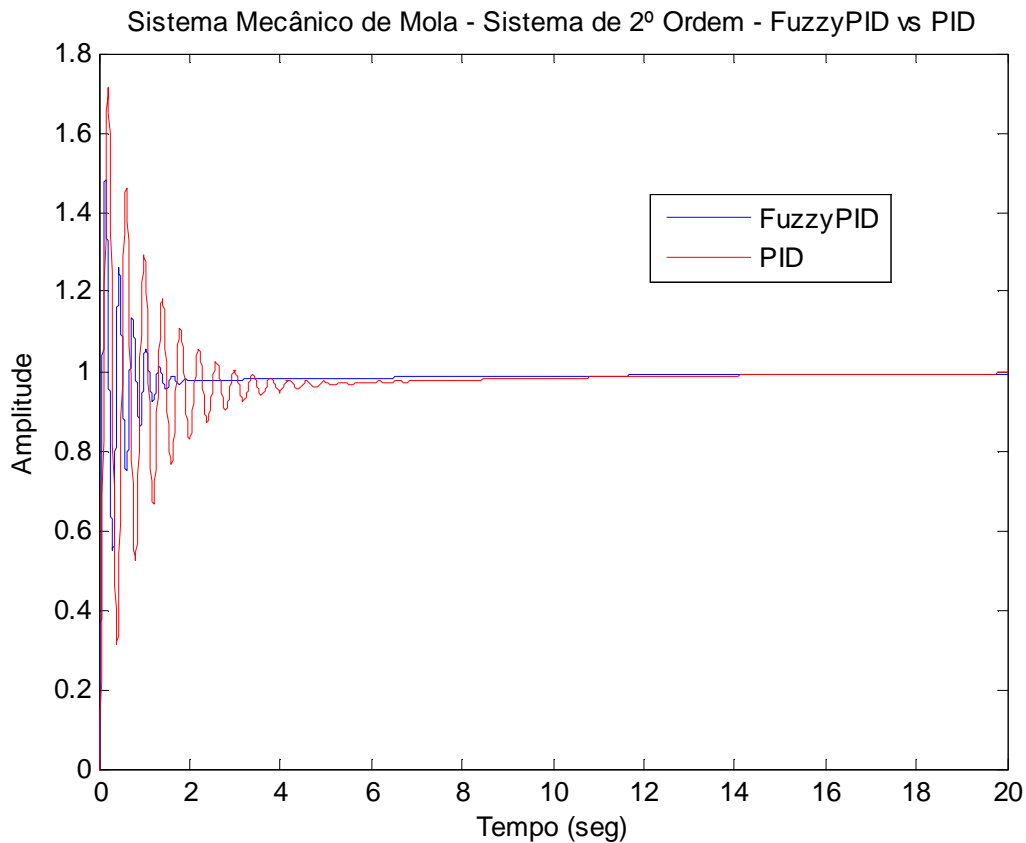


Figura 59 - Análise temporal de um sistema mecânico de uma mola

Para o sistema mecânico de uma mola a avaliação em termos de tempo de subida complica-se devido às oscilações patentes nos sinais de ambos os controladores. Contudo, consegue-se observar que o FuzzyPID apresenta menores oscilações que o PID e que o seu tempo de estabelecimento é substancialmente menor.

Tabela 15 - Valores dos parâmetros temporais do sistema mecânico de uma mola

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 50]	[-10 15]	[-1 1]	2s
PID	15,43	2	0,04	8s

6.1.4. SISTEMA ELÉCTRICO RLC SÉRIE

Um sistema eléctrico RLC série também é descrito por um sistema de segunda ordem, sendo representado pela seguinte função de transferência:

$$TF = \frac{1}{0,04s^2 + 0,6s + 1}$$

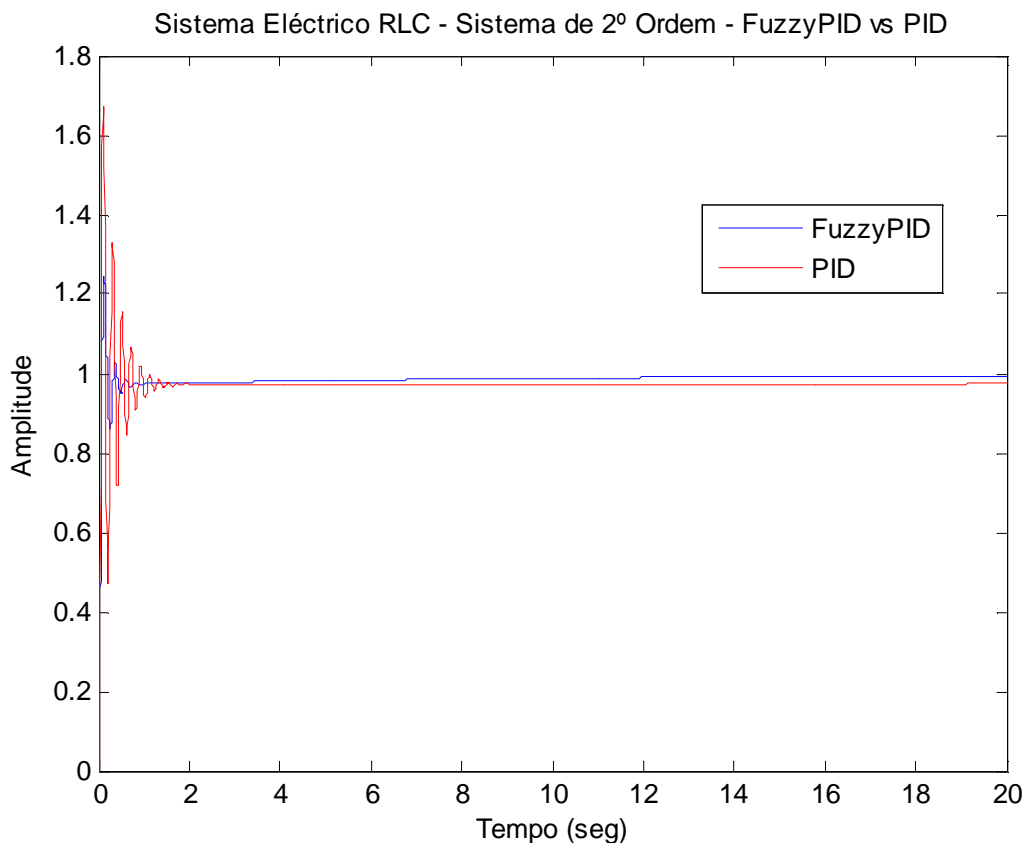


Figura 60 - Análise temporal do sistema eléctrico RLC

Para o sistema eléctrico RLC os resultados obtidos são muito similares ao sistema anterior. Nos primeiros segundos são observáveis algumas oscilações nas respostas dos controladores, contudo é de referir que para o FuzzyPID essas mesmas oscilações são

bastante menores, o que indica maior capacidade de estabilização. Ambos os controladores fornecem uma boa resposta em termos de tempo de estabelecimento.

Tabela 16 - Valores dos parâmetros temporais do sistema eléctrico RLC Série

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 50]	[-10 15]	[-1 1]	0,9s
PID	35,4	0,08	0,02	1,6s

6.1.5. SISTEMA DE CONTROLO DE POSIÇÃO MOTOR DC

Este sistema de controlo de posição de um motor DC também é descrito por uma função de transferência de 2ª ordem dada por:

$$TF = \frac{9}{0,05s^2 + 0,51s + 9,1}$$

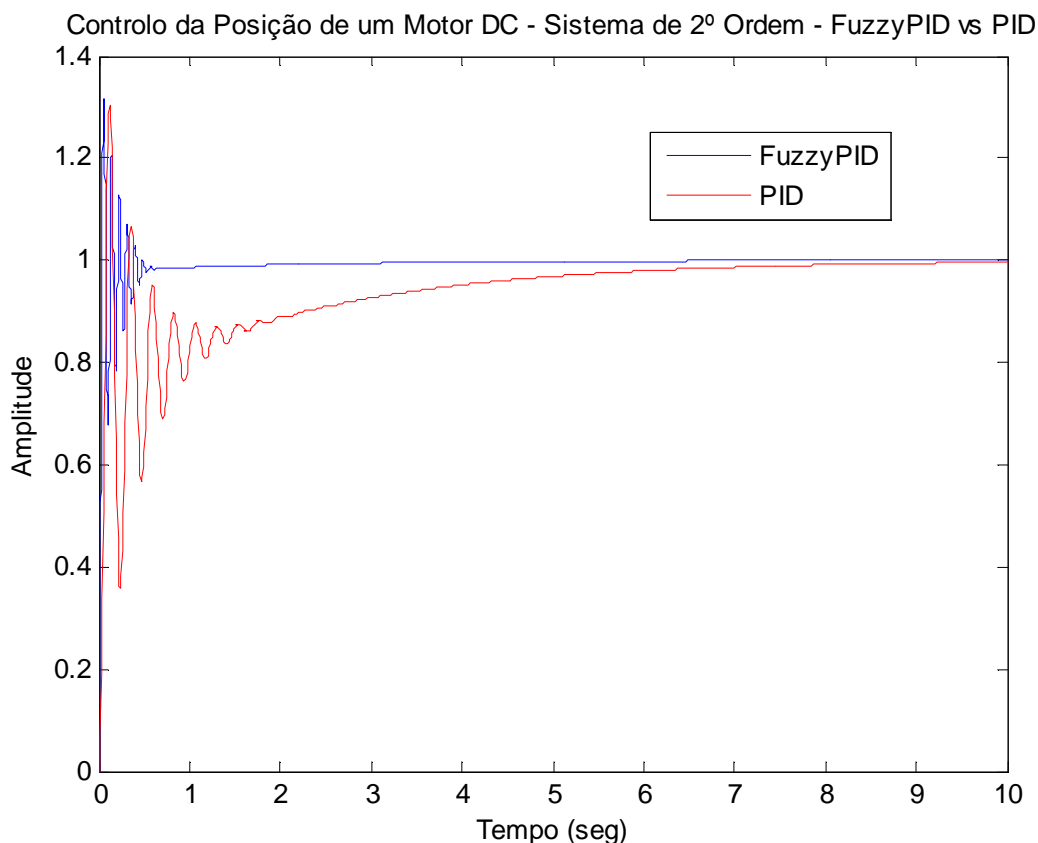


Figura 61 - Análise temporal do sistema de controlo da posição de um motor DC

Para este sistema, pode-se observar diversas oscilações iniciais, tanto para o controlador FuzzyPID como para o PID, pelo que, este fenómeno irá influenciar o tempo de subida de

ambos. Para o controlador FuzzyPID tem-se um tempo de estabelecimento de aproximadamente 3 segundos enquanto para o PID fixa-se nos 9,5 segundos.

Tabela 17 - Valores dos parâmetros temporais do sistema de controlo da posição de um motor DC

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 50]	[-10 40]	[-1 1]	3s
PID	2,79	1,6	0,0125	9,5s

6.1.6. SISTEMA DE CONTROLO DE VELOCIDADE MOTOR DC

Este sistema é descrito pela seguinte função de transferência de 2ª ordem:

$$TF = \frac{0,01}{0,005s^2 + 0,06s + 0,1001}$$

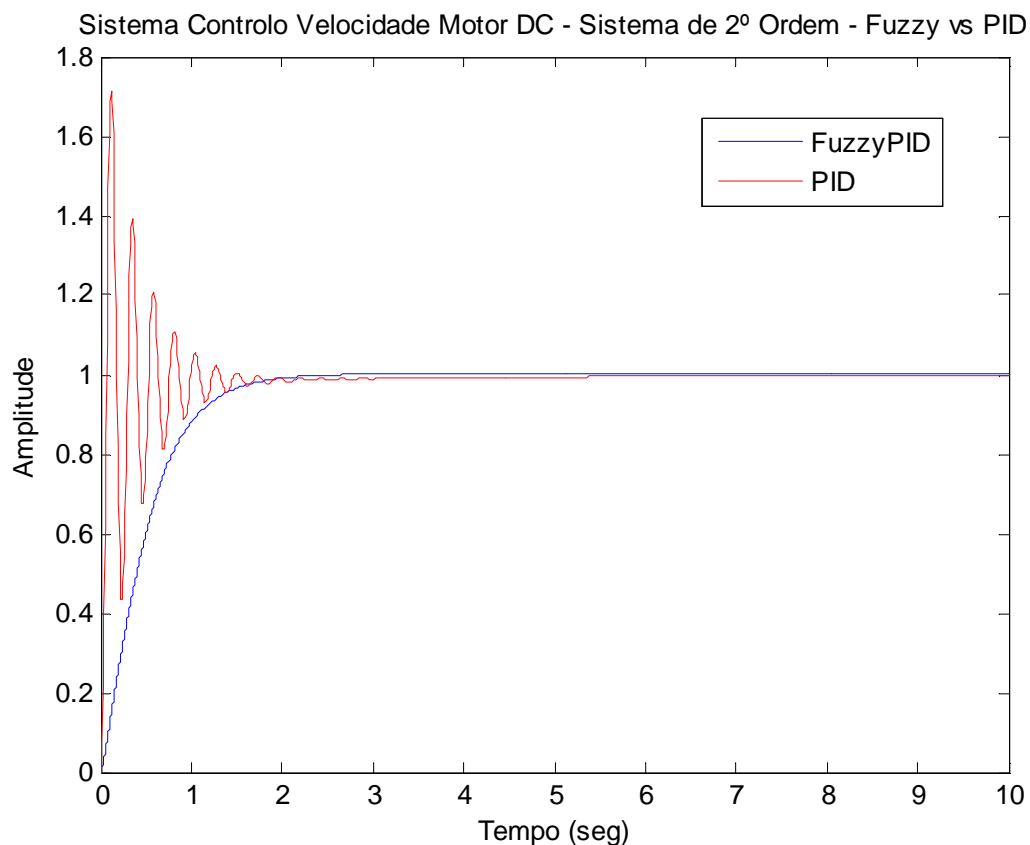


Figura 62 - Análise temporal do sistema de controlo da velocidade de um motor DC

Para o sistema de Controlo da Velocidade de um Motor DC, é bem clara a vantagem na utilização do controlador FuzzyPID. Olhando para o tempo de subida obtém-se cerca de 1,5 segundos para o FuzzyPID, enquanto para o PID o sistema entra numa oscilação

convergente, onde torna-se difícil obter o valor para o tempo de subida. Relativamente ao *overshoot*, este não existe para o FuzzyPID, e o tempo de estabelecimento fixa-se em 2 segundos, muito abaixo do obtido para o PID.

Tabela 18 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 50]	[0 40]	[0 15]	2s
PID	352,5	0,9	0,225	10s

6.1.7. PRIMEIRO SISTEMA DE TERCEIRA ORDEM

Nesta subsecção apresenta-se um exemplo de uma função de transferência de terceira ordem, dada por:

$$TF = \frac{1}{s^3 + 9s^2 + 26s + 24}$$

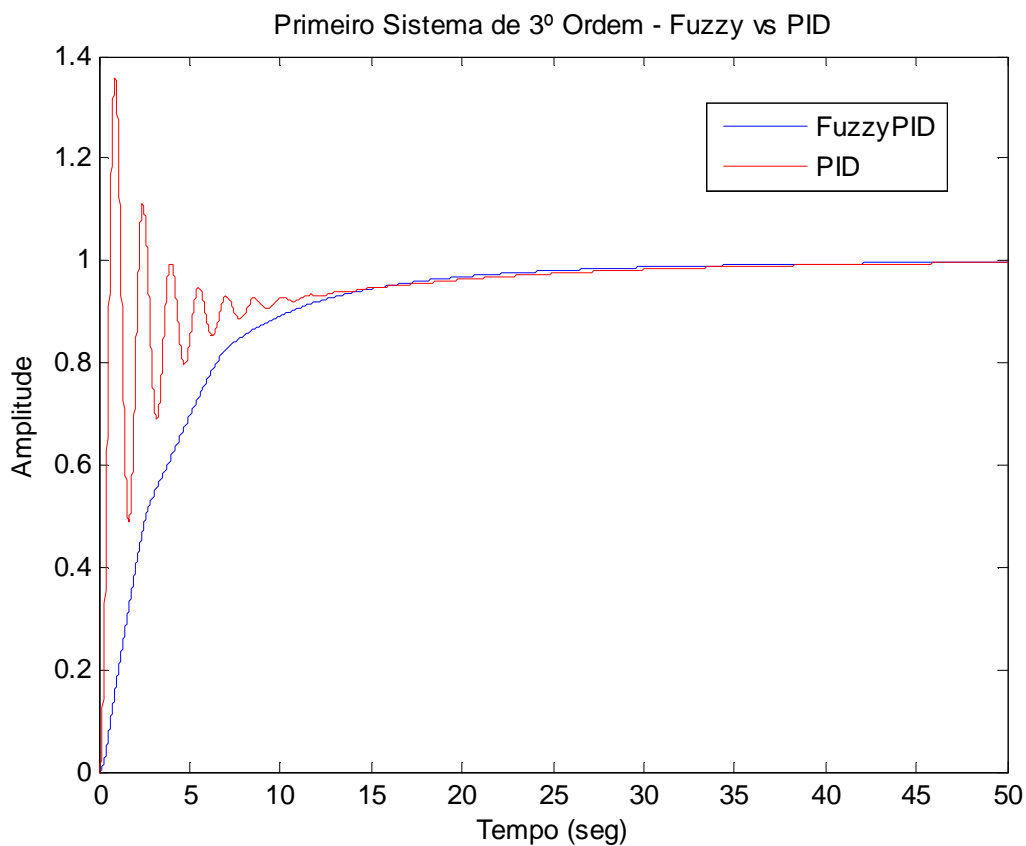


Figura 63 - Análise temporal do primeiro sistema de 3ª ordem

Para o sistema de 3ª ordem apresentado acima, observa-se uma situação semelhante à anterior, onde se verifica novamente uma grande vantagem na utilização de um controlador FuzzyPID. De facto, para o FuzzyPID o tempo de subida é de 12,5 segundos, não apresenta *overshoot* e o seu tempo de estabelecimento fixa-se em 30 segundos. Para o controlador PID este apresenta algumas oscilações iniciais, e quando estabilizado o sinal de saída acompanha o sinal do controlador FuzzyPID. Desta forma o tempo de estabelecimento mantém-se igual para ambos os controladores.

Tabela 19 - Valores dos parâmetros temporais do primeiro sistema de 3ª ordem

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 10]	[0 10]	[-1 1]	30s
PID	110,25	10	0,156	30s

6.1.8. SEGUNDO SISTEMA DE TERCEIRA ORDEM

Outro dos sistemas estudados é descrito pela seguinte função de transferência de 3ª ordem:

$$TF = \frac{8}{s^3 + 11,5s^2 + 14s + 8}$$

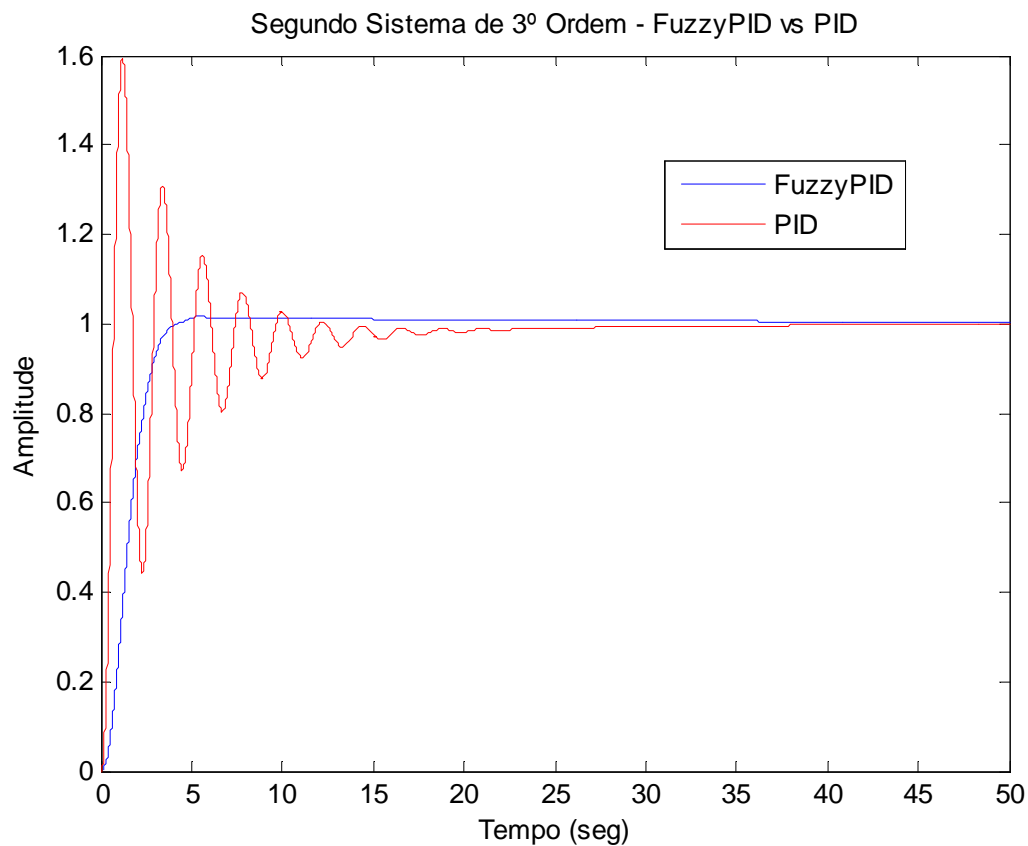


Figura 64 - Análise temporal do segundo sistema de 3ª ordem

Para o sistema de terceira ordem apresentado, obteve-se uma resposta distinta entre os controladores utilizados. Para o FuzzyPID tem-se um tempo de subida de 3,5 segundos, sendo o *overshoot* assumido como nulo e o tempo de estabelecimento de 6,5 segundos. Para o controlador PID o sistema apresenta algumas oscilações nos segundos iniciais, demorando cerca de 20 segundos a estabilizar, pelo que o seu tempo de estabelecimento fixa-se nos 22,5 segundos.

Tabela 20 - Valores dos parâmetros temporais do segundo sistema de 3ª ordem

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-1 1]	[-10 10]	[-1 1]	6,5s
PID	10,2	0,9	0,23	22,5s

6.1.9. TERCEIRO SISTEMA DE TERCEIRA ORDEM

Este sistema é descrito pela seguinte função de transferência, também ela de 3ª ordem dada por:

$$TF = \frac{1}{s^3 + 7s^2 + 11s + 5}$$

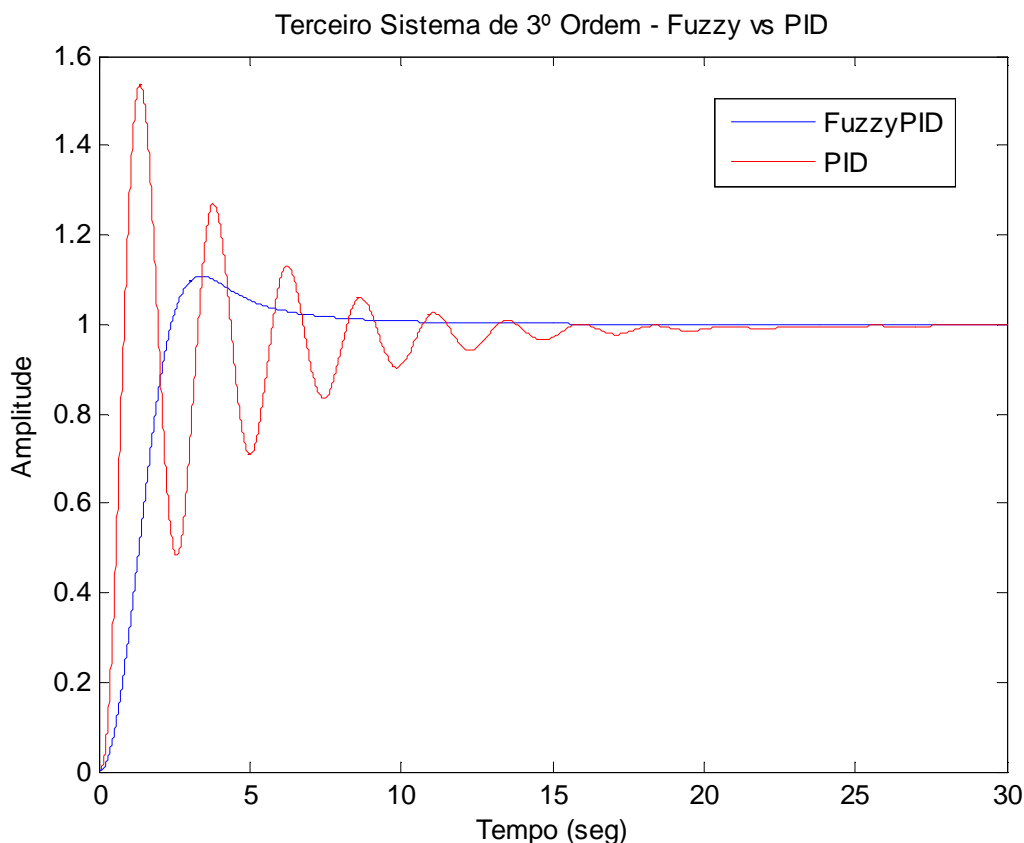


Figura 65 - Análise temporal do terceiro sistema de 3ª ordem

Relativamente ao terceiro sistema de terceira ordem, último sistema sem atraso simulado, o resultado apresentado não diverge dos resultados obtidos anteriormente. Para o controlador FuzzyPID o tempo de subida situa-se nos 3 segundos, apresenta um ligeiro *overshoot* na ordem dos 10% do valor de entrada, e o tempo de estabelecimento fixa-se nos 10 segundos. Relativamente ao controlador PID, este apresenta algumas oscilações iniciais, que acabam por anular-se após o segundo 20, sendo este o tempo de estabelecimento.

Tabela 21 - Valores dos parâmetros do terceiro sistema de 3ª ordem

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[0 10]	[0 10]	[-1 1]	10s
PID	38,7	0,9	0,23	20s

6.2. SISTEMAS COM ATRASO

6.2.1. PRIMEIRO SISTEMA DE 1º ORDEM COM ATRASO 5s

O sistema de 1ª ordem com atraso inicialmente estudado, foi efectuado tendo por base a seguinte função de transferência onde é patente um atraso de 5 segundos:

$$TF = \frac{1}{s + 1} e^{-5s}$$

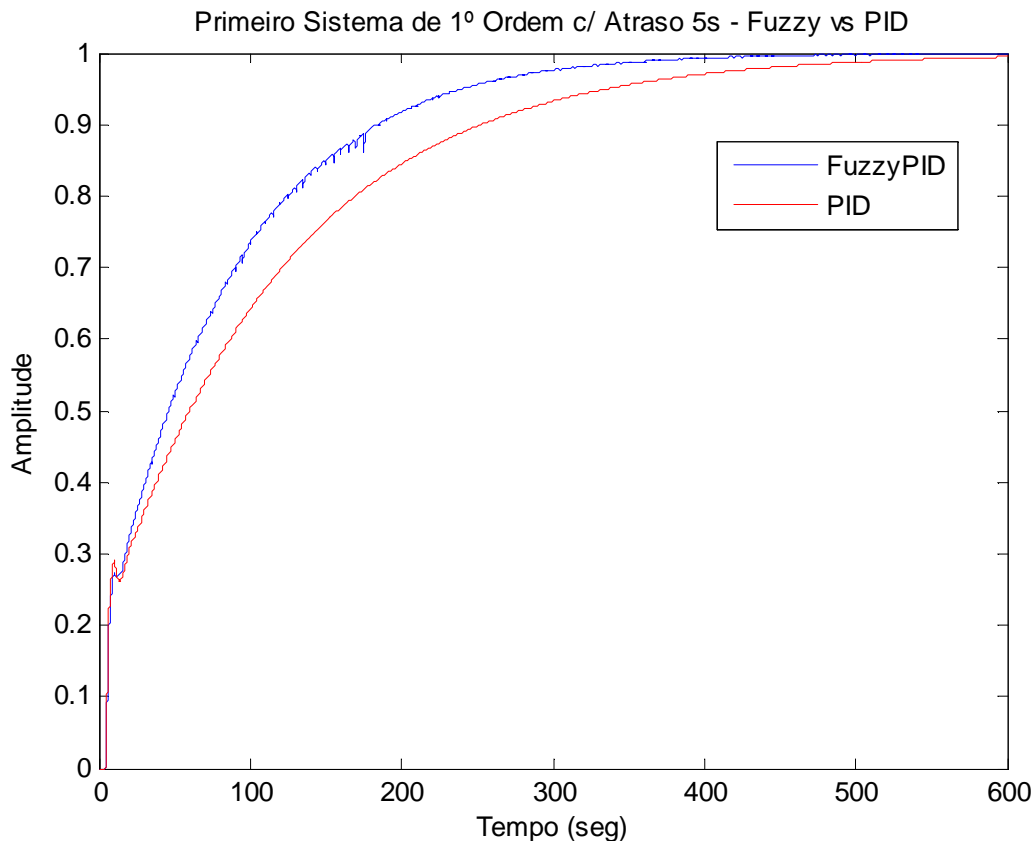


Figura 66 - Análise temporal do primeiro sistema de 1ª ordem com atraso de 5s

Em relação ao primeiro sistema de 1ª ordem com atraso de 5 segundos podemos constatar que para ambos os controladores, FuzzyPID e PID, o sistema não apresenta *overshoot*, contudo para o controlador FuzzyPID o sistema em causa apresenta um menor tempo de subida e um menor tempo de estabelecimento, o que o torna melhor quando comparado com o PID clássico. De referir, que os sistemas com atraso na sua resposta são mais difíceis de controlar, o que evidencia a importância do controlador FuzzyPID.

Tabela 22 - Valores dos parâmetros temporais para o primeiro sistema de 1ª ordem com atraso de 5s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,65 0,65]	[-0,04 0,04]	[-0,6 0,6]	350s
PID	0,25	0,01	0,01	525s

6.2.2. PRIMEIRO SISTEMA DE 1ª ORDEM COM ATRASO 30s

O mesmo sistema estudado na subsecção anterior mas agora com 30 segundos de atraso é representado por:

$$TF = \frac{1}{s + 1} e^{-30s}$$

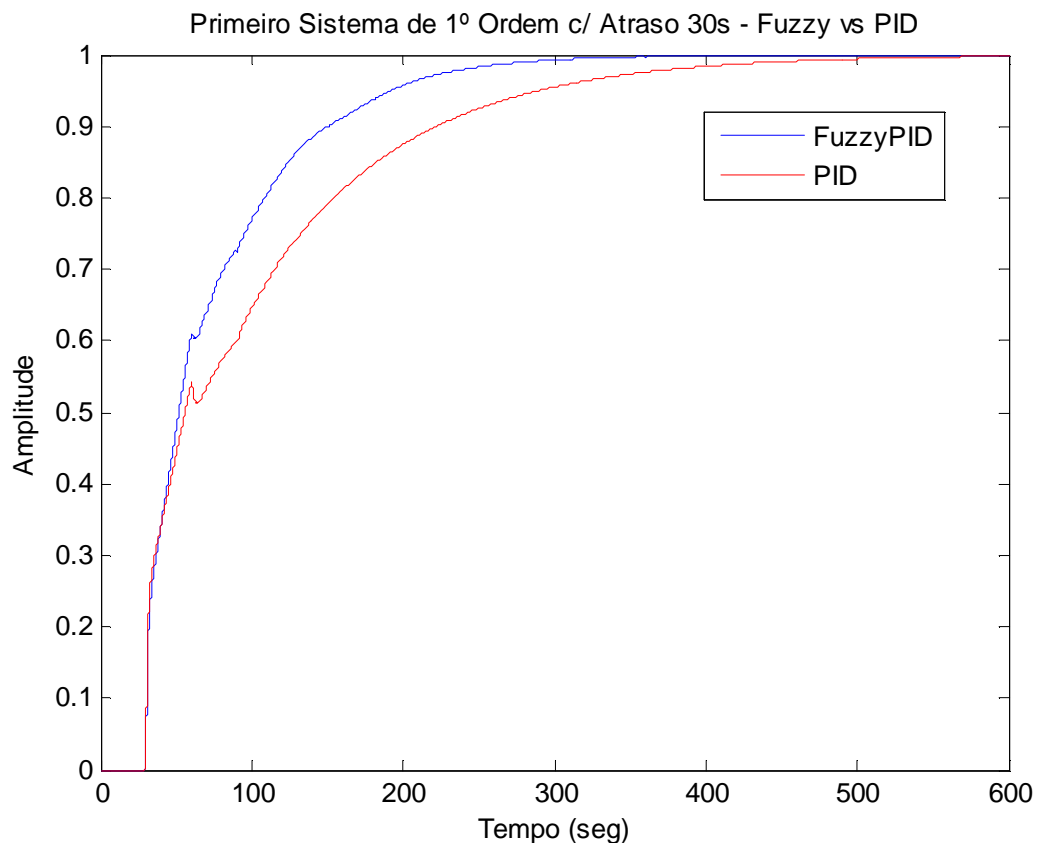


Figura 67 - Análise temporal do primeiro sistema de 1ª ordem com atraso de 30s

Para o mesmo sistema de 1ª ordem, alterando o atraso de 5 para 30 segundos, observa-se que se mantém o padrão de resposta, pelo que o controlador FuzzyPID continua a ser a primeira opção. A resposta não apresenta *overshoot*, os tempos de subida e de estabelecimento continuam a ser inferiores para o controlador FuzzyPID.

Tabela 23 - Valores dos parâmetros temporais do primeiro sistema de 1ª ordem com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,65 0,65]	[-0,04 0,04]	[-0,6 0,6]	300s
PID	0,25	0,01	0,01	500s

6.2.3. SEGUNDO SISTEMA DE 1ª ORDEM COM ATRASO 5s

Um segundo sistema de 1ª ordem com atraso é o representado pela seguinte função de transferência:

$$TF = \frac{14}{4s + 14} e^{-5s}$$

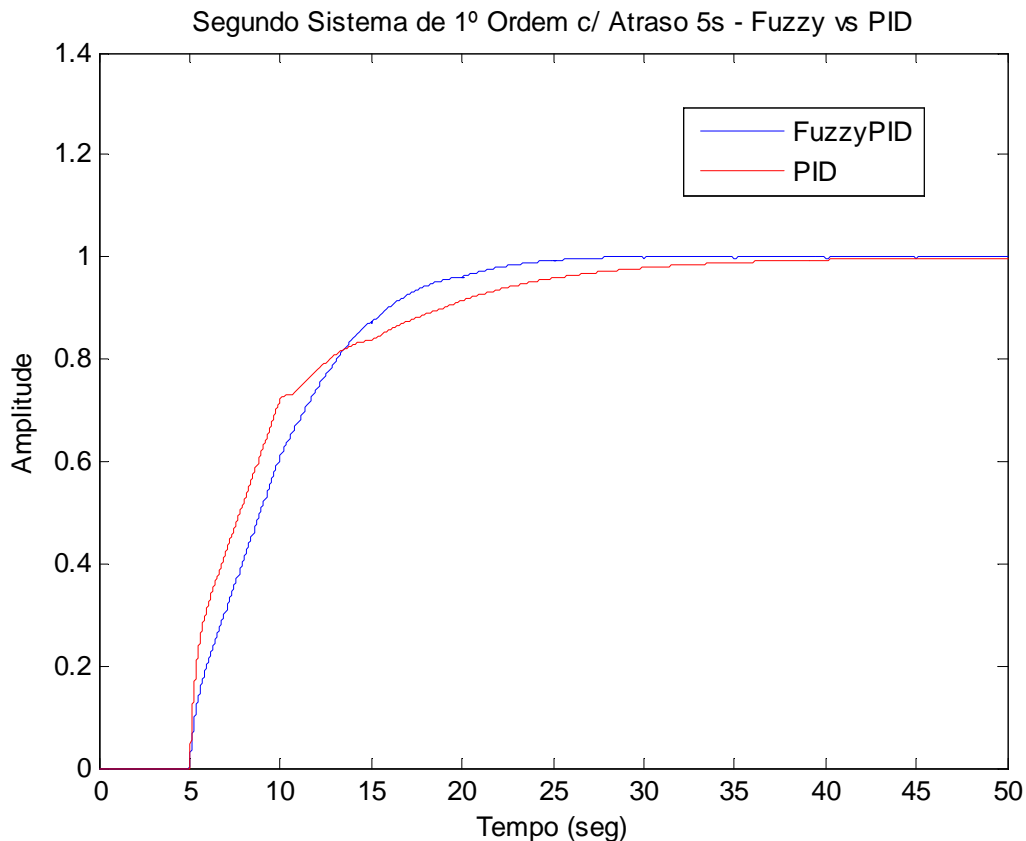


Figura 68 - Análise temporal do segundo sistema de 1ª ordem com atraso de 5s

Neste caso, o sistema apresenta um atraso de 5 segundos revelando uma resposta similar aos dois sistemas anteriores. Entre os segundos 5 e 10 o controlador PID apresenta uma resposta mais rápida, o que fazia crer um tempo de subida inferior ao FuzzyPID, contudo no final isso não se verificou pelo que se pode concluir que o controlador FuzzyPID é mais

rápido a reagir. Posto isto, pode-se afirmar que o controlador FuzzyPID apresenta um menor tempo de subida, não apresenta *overshoot* e revela um menor tempo de estabelecimento.

Tabela 24 - Valores dos parâmetros temporais do segundo sistema de 1ª ordem com atraso de 5s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,65 0,65]	[-0,3 0,3]	[-0,6 0,6]	25s
PID	0,25	0,1	0,01	35s

6.2.4. SEGUNDO SISTEMA DE 1ª ORDEM COM ATRASO 30S

O mesmo sistema apresentado anteriormente mas agora com um atraso superior, de 30 segundos, é representado por:

$$TF = \frac{14}{4s + 14} e^{-30s}$$

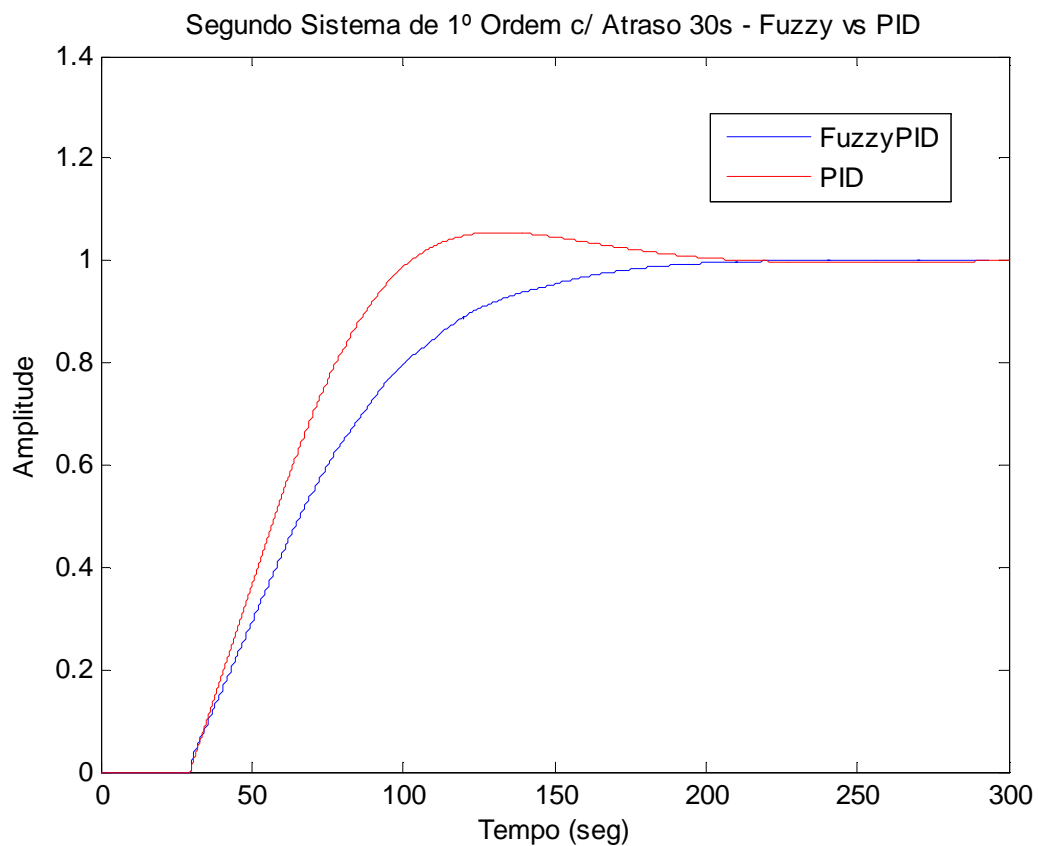


Figura 69 - Análise temporal do segundo sistema de 1ª ordem com atraso de 30s

Para este sistema de 1ª ordem com atraso de 30 segundos pode-se observar que o controlador PID apresenta um menor tempo de subida. Mas devido ao declive que permite um tempo de subida inferior, a sua resposta fica caracterizada por conter *overshoot* o que em alguns sistemas pode ser prejudicial ao controlo. Por outro lado, o controlador FuzzyPID, mesmo sendo um pouco mais lento na resposta transitória, não apresenta *overshoot* e tende a estabilizar mais cedo, em cerca de 20 segundos.

Tabela 25 - Valores dos parâmetros temporais do segundo sistema de 1ª ordem com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,4 0,4]	[-0,04 0,04]	[-0,6 0,6]	180s
PID	0,25	0,01	0,01	200s

6.2.5. TERCEIRO SISTEMA DE 1ª ORDEM COM ATRASO 30s

Outro sistema de 1ª ordem com atraso estudado é dado pela seguinte função de transferência:

$$TF = \frac{34}{300s + 54} e^{-30s}$$

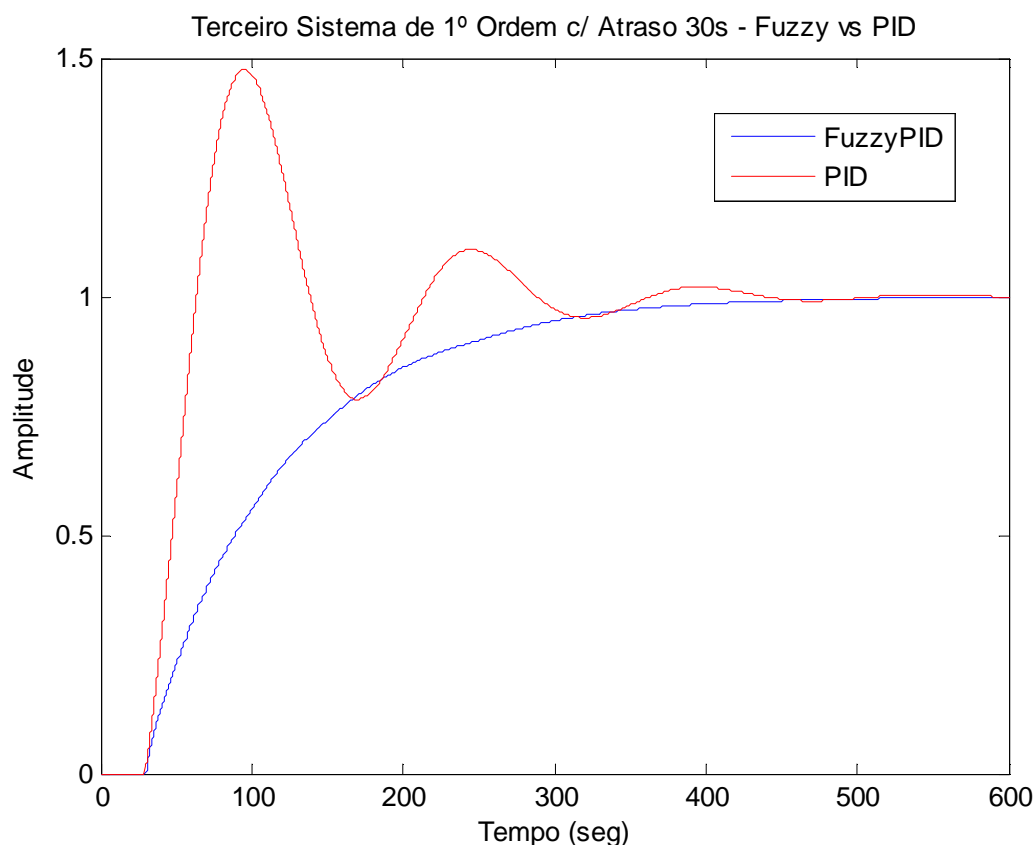


Figura 70 - Análise temporal do terceiro sistema de 1ª ordem com atraso de 30s

Relativamente a este sistema de 1ª ordem com um atraso de 30 segundos, facilmente se conclui que para o controlador FuzzyPID a resposta é muito melhor. Para o controlador FuzzyPID a resposta do sistema é estável, não apresenta oscilações indesejadas e o seu tempo de estabelecimento é inferior ao do PID. O controlador PID apresenta no entanto um tempo de subida significativamente inferior, no entanto, revela um acentuado *overshoot* o qual será sempre desvantajoso para qualquer sistema a controlar.

Tabela 26 - Valores dos parâmetros temporais do terceiro sistema de 1ª ordem com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,65 0,65]	[-0,04 0,04]	[-0,6 0,6]	450s
PID	0,75	0,075	0,01	500s

6.2.6. SISTEMA MECÂNICO DE UMA MOLA DE 2ª ORDEM COM ATRASO 10s

A função de transferência que descreve um sistema mecânico de uma mola é dada por:

$$TF = \frac{14,9691}{s^2 + 4s + 14,9691} e^{-10s}$$

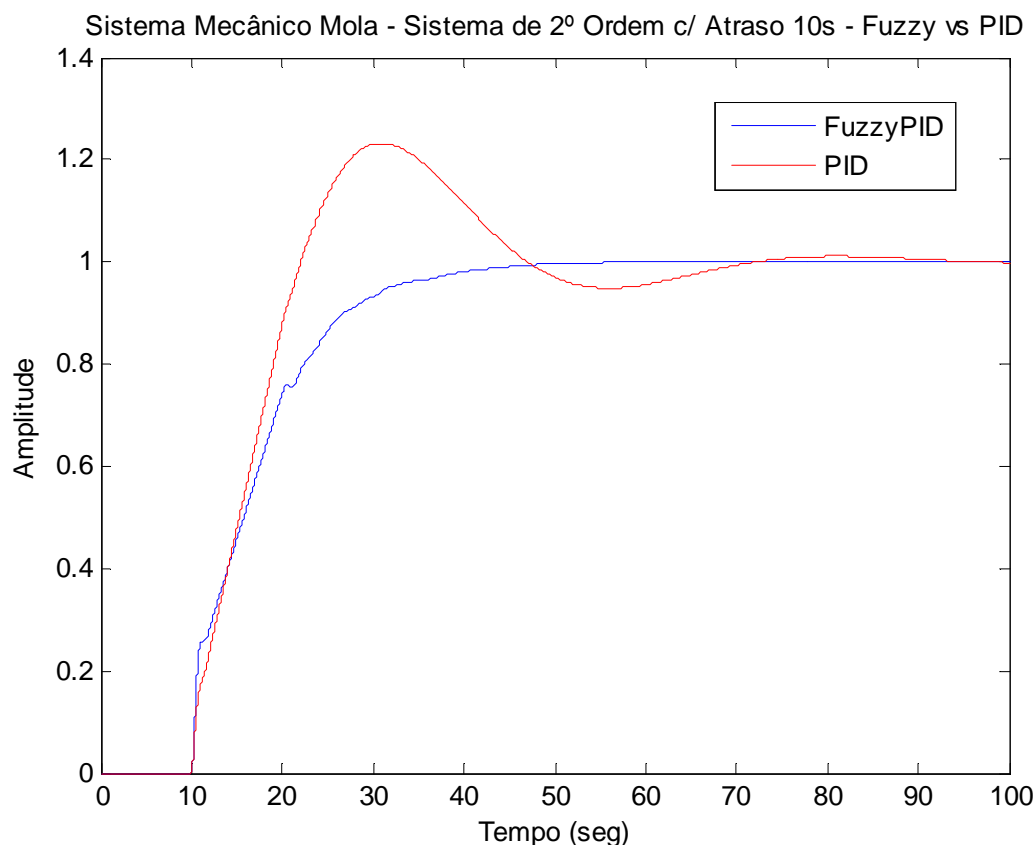


Figura 71 - Análise temporal do sistema mecânico de uma mola com atraso de 10s

Para o sistema mecânico de uma mola com atraso de 10 segundos podemos constatar que a resposta do controlador PID é caracterizada por algumas oscilações e por um tempo de subida inferior ao controlador FuzzyPID. Por outro lado, o controlador FuzzyPID não apresenta *overshoot* e o seu tempo de estabelecimento é substancialmente inferior ao do PID.

Tabela 27 - Valores dos parâmetros temporais do sistema mecânico de uma mola com atraso de 10s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,55 0,55]	[-0,17 0,17]	[-0,6 0,6]	40s
PID	0,1	0,08	0,01	80s

6.2.7. SISTEMA MECÂNICO DE UMA MOLA DE 2º ORDEM COM ATRASO 30s

O mesmo sistema da subsecção anterior, mas agora com 30 segundos de atraso é dada por:

$$TF = \frac{14,9691}{s^2 + 4s + 14,9691} e^{-30s}$$

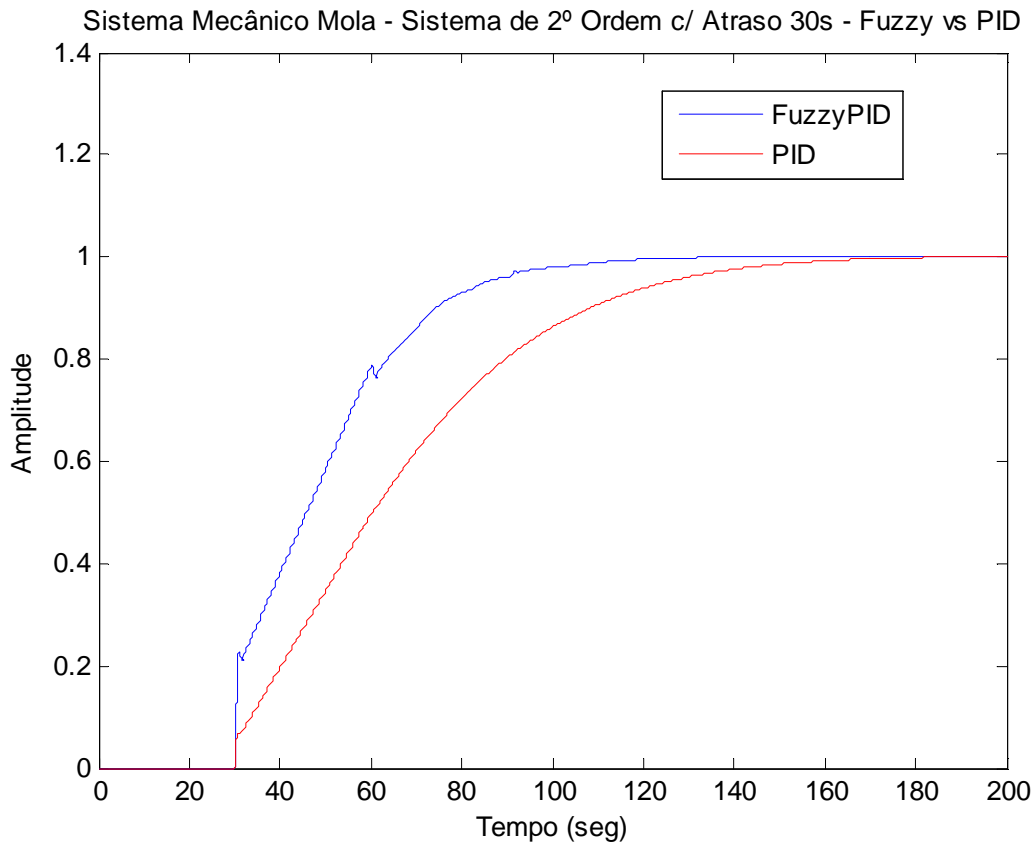


Figura 72 - Análise temporal do sistema mecânico de uma mola com atraso de 30s

Para o sistema mecânico mola com atraso de 30 segundos a resposta obtida diverge bastante da resposta do mesmo sistema com atraso de 10 segundos. Neste caso o controlador FuzzyPID apresenta um menor tempo de subida e um menor tempo de estabelecimento, ambos os controladores não apresentam *overshoot*.

Mais uma vez, se verifica a vantagem no uso do FuzzyPID, pois, mesmo quando o sistema revela tempos de atraso elevados, o controlador consegue reagir bem.

Tabela 28 - Valores dos parâmetros temporais do sistema mecânico de uma mola com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,55 0,55]	[-0,06 0,06]	[-0,6 0,6]	80s
PID	0,1	0,03	0,01	150s

6.2.8. SISTEMA ELÉCTRICO RLC DE 2ª ORDEM COM ATRASO 30s

Nesta subsecção apresenta-se um sistema eléctrico RLC série descrito por uma função de transferência de 2ª ordem com atraso de 30 segundo dada por:

$$TF = \frac{1}{0,04s^2 + 0,6s + 1} e^{-30s}$$

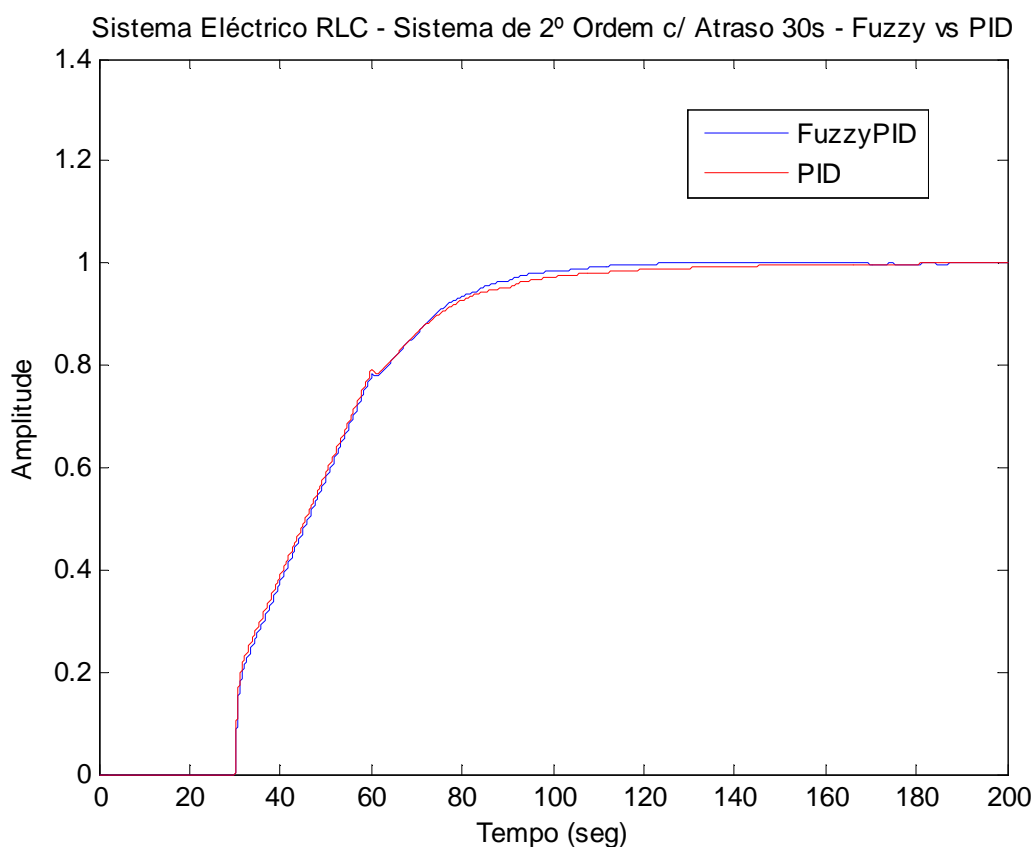


Figura 73 - Análise temporal do sistema eléctrico RLC série com atraso de 30s

Para o sistema eléctrico RLC com atraso de 30 segundos, constata-se que a resposta de ambos os controladores, FuzzyPID e PID, são idênticas. O tempo de subida é igual para ambos, não apresentam *overshoot*, contudo para o tempo de estabelecimento o controlador FuzzyPID tem tendência a estabilizar mais cedo, sendo essa diferença de aproximadamente 20 segundos.

Tabela 29 - Valores dos parâmetros temporais do sistema eléctrico RLC série com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,55 0,55]	[-0,06 0,06]	[-0,6 0,6]	110s
PID	0,2	0,02	0,01	130s

6.2.9. SISTEMA DE CONTROLO DA POSIÇÃO DE UM MOTOR DC – 2ª ORDEM COM ATRASO 30s

A função de transferência que caracteriza este sistema é dada por:

$$TF = \frac{9}{0,05s^2 + 0,51s + 9,1} e^{-30s}$$

Sistema Controlo Posição Motor DC - Sistema de 2º Ordem c/ Atraso 30s - Fuzzy vs PID

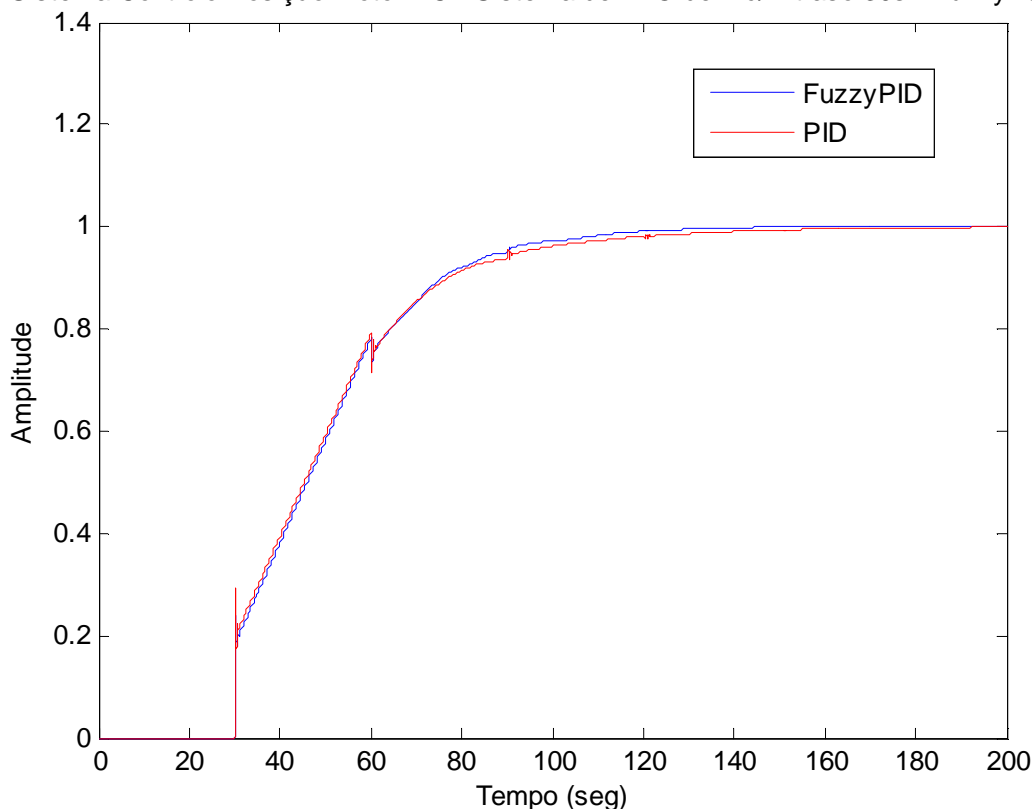


Figura 74 - Análise temporal do sistema de controlo da posição de um motor DC com atraso de 30s

Para o sistema de controlo da posição de um motor DC com atraso de 30 segundos, o tipo de resposta assemelha-se ao sistema anterior. Ambos os controladores apresentam desempenhos idênticos no diz respeito ao tempo de subida e *overshoot*, contudo para o tempo de estabelecimento apresentam uma pequena diferença de 10 segundos.

Tabela 30 - valores dos parâmetros temporais do sistema de controlo da posição de um motor DC com atraso de 30s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,55 0,55]	[-0,06 0,06]	[-0,6 0,6]	110s
PID	0,2	0,02	0,01	120s

6.2.10. SISTEMA DE CONTROLO DA VELOCIDADE DE UM MOTOR DC – 2ª ORDEM COM ATRASO 10S

Outro sistema de controlo com atraso é descrito na seguinte função de transferência:

$$TF = \frac{0,01}{0,005s^2 + 0,06s + 0,1001} e^{-10s}$$

Sistema Controlo Velocidade Motor DC - Sistema de 2º Ordem c/ Atraso 10s - Fuzzy vs PID

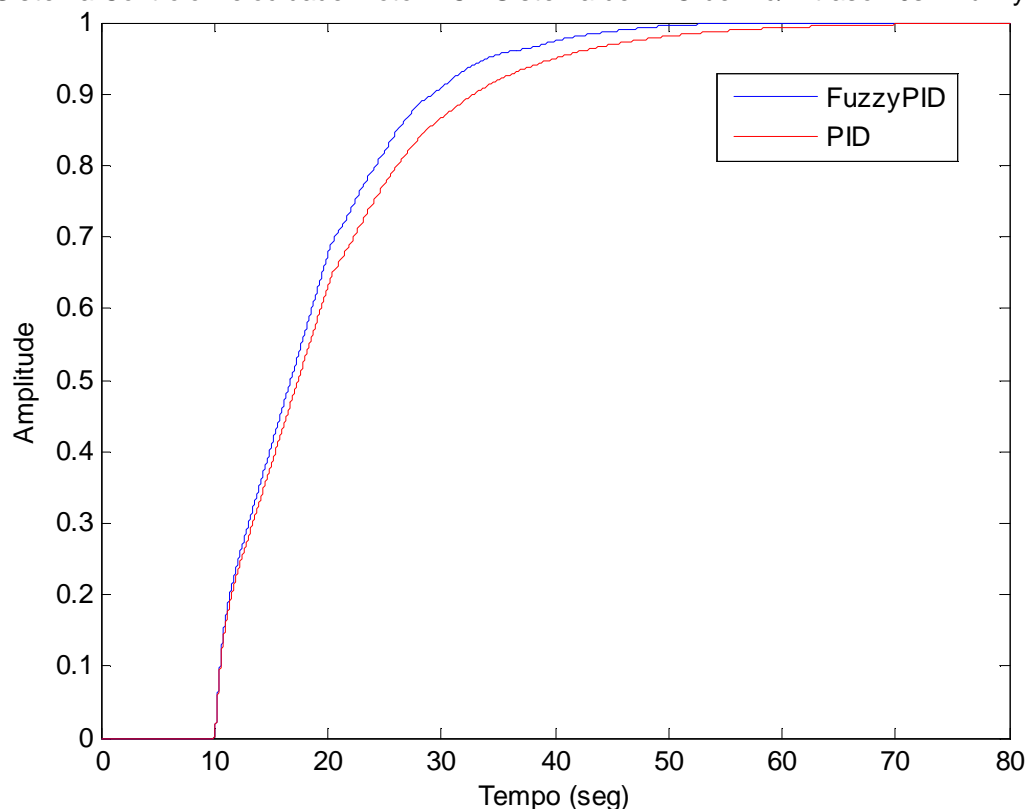


Figura 75 - Análise temporal do sistema de controlo da velocidade de um motor DC com atraso de 10s

Para o sistema de controlo da velocidade de um motor DC com um atraso de 10 segundos podemos observar que o controlador FuzzyPID apresenta um menor tempo de subida e um menor tempo de estabelecimento, fixado em 50 segundos. Ambos os controladores não apresentam *overshoot*.

Tabela 31 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC com atraso de 10s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,5 0,5]	[-1,6 1,6]	[-0,6 0,6]	50s
PID	1	0,45	0,01	65s

6.2.11. SISTEMA DE CONTROLO DA VELOCIDADE DE UM MOTOR DC – 2ª ORDEM COM ATRASO 30s

O controlo da velocidade de um motor DC com um atraso de 30 segundos é descrito pela seguinte função de transferência:

$$TF = \frac{0,01}{0,005s^2 + 0,06s + 0,1001} e^{-30s}$$

Sistema Controlo Velocidade Motor DC - Sistema de 2º Ordem c/ Atraso 30s - Fuzzy vs PID

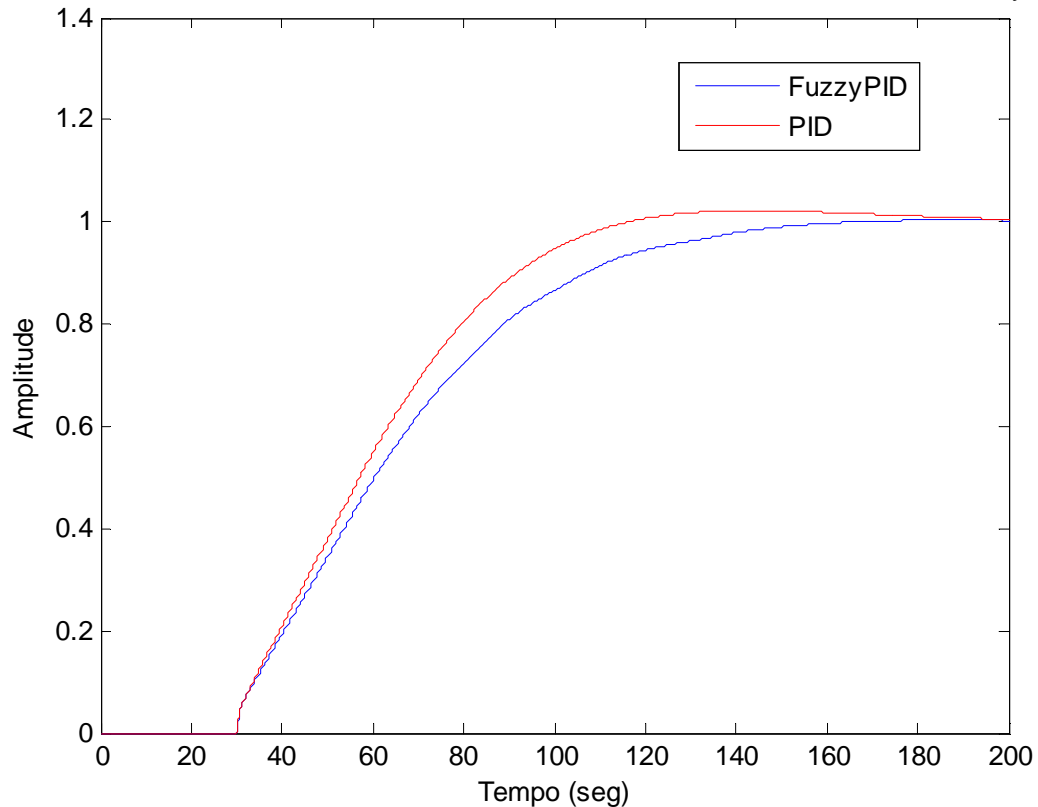


Figura 76 - Análise temporal do sistema de controlo da velocidade de um motor DC com atraso de 10s

Para este sistema observou-se que o controlador PID apresenta um menor tempo de subida. Por outro lado o controlador FuzzyPID não apresenta *overshoot* e o seu tempo de estabelecimento é inferior. Neste caso o controlador PID é mais rápido a atingir o *setpoint* mas mais lento a estabilizar.

Tabela 32 - Valores dos parâmetros temporais do sistema de controlo da velocidade de um motor DC com atraso de 10s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-1,5 1,5]	[-0,45 0,45]	[-0,6 0,6]	50s
PID	0,5	0,17	0,01	65s

6.2.12. PRIMEIRO SISTEMA DE TERCEIRA ORDEM COM ATRASO 50s

Nesta subsecção vai ser apresentada uma função de transferência característica de um sistema de 3ª ordem com atraso de 50 segundos. Devido ao facto de os sistemas de ordem superior poderem ser descritos por vários sistemas de ordens inferiores, não se conseguiu obter funções de transferência características de sistemas reais, daí se terem utilizado para este estudo, expressões de 3ª ordem clássicas.

$$TF = \frac{7,98}{s^3 + 11,4s^2 + 14s + 7,98} e^{-50s}$$

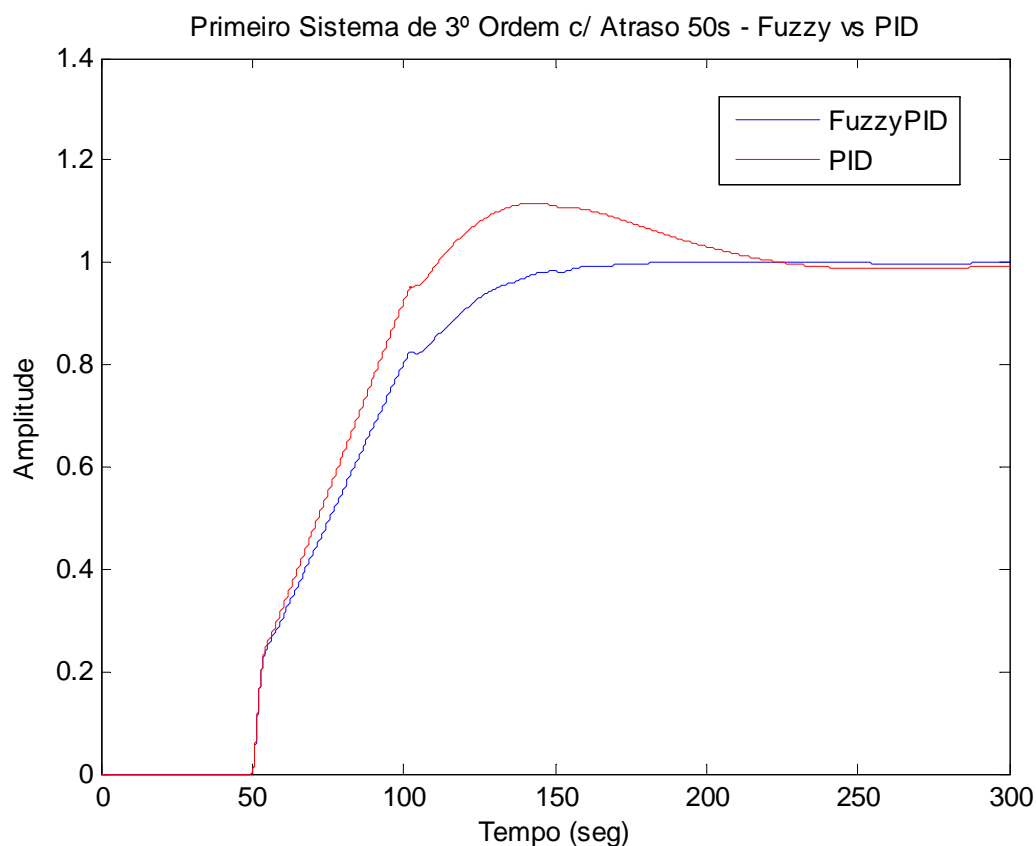


Figura 77 - Análise temporal do primeiro sistema de 3ª ordem com atraso de 50s

Pela análise do sistema aqui apresentado, pode-se concluir que o controlador FuzzyPID apresenta um tempo de subida ligeiramente superior ao PID, contudo não apresenta *overshoot* e o seu tempo de estabelecimento é substancialmente inferior, fixando-se em 170 segundos.

Tabela 33 - Valores dos parâmetros temporais do primeiro sistema de 3ª ordem com atraso de 50s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,6 0,6]	[-0,037 0,037]	[-0,6 0,6]	170s
PID	0,2	0,015	0,01	250s

6.2.13. SEGUNDO SISTEMA DE TERCEIRA ORDEM COM ATRASO 50s

Outro dos sistemas de 3ª ordem com atraso estudado é o representado pela seguinte função de transferência, onde o atraso considerado é de 50 segundos:

$$TF = \frac{10}{s^3 + 11s^2 + 11s + 10} e^{-50s}$$

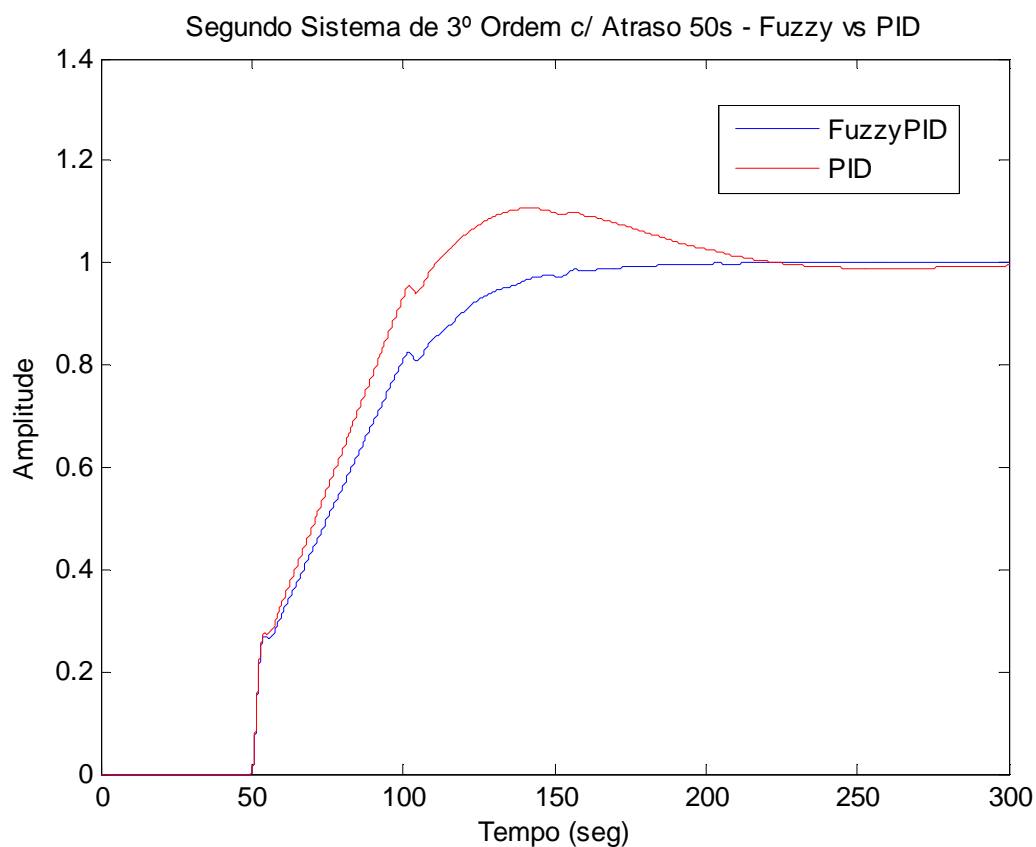


Figura 78 - Análise temporal do segundo sistema de 3ª ordem com atraso de 50s

A resposta dos controladores FuzzyPID e PID para o sistema aqui estudado é bastante semelhante ao sistema anterior. Apesar do controlador PID apresentar um tempo de subida ligeiramente inferior, o controlador FuzzyPID não apresenta *overshoot* e o seu tempo de estabelecimento situa-se nos 190 segundos, sendo muito inferior ao do PID.

Tabela 34 - Valores dos parâmetros temporais do segundo sistema de 3ª ordem com atraso de 50s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,6 0,6]	[-0,037 0,037]	[-0,6 0,6]	190s
PID	0,2	0,015	0,01	250s

6.2.14. TERCEIRO SISTEMA DE TERCEIRA ORDEM COM ATRASO 50s

Nesta subsecção apresentamos outro sistema de 3ª ordem com atraso de 50 segundos, descrito pela seguinte função de transferência:

$$TF = \frac{1}{s^3 + 9s^2 + 26s + 24} e^{50s}$$

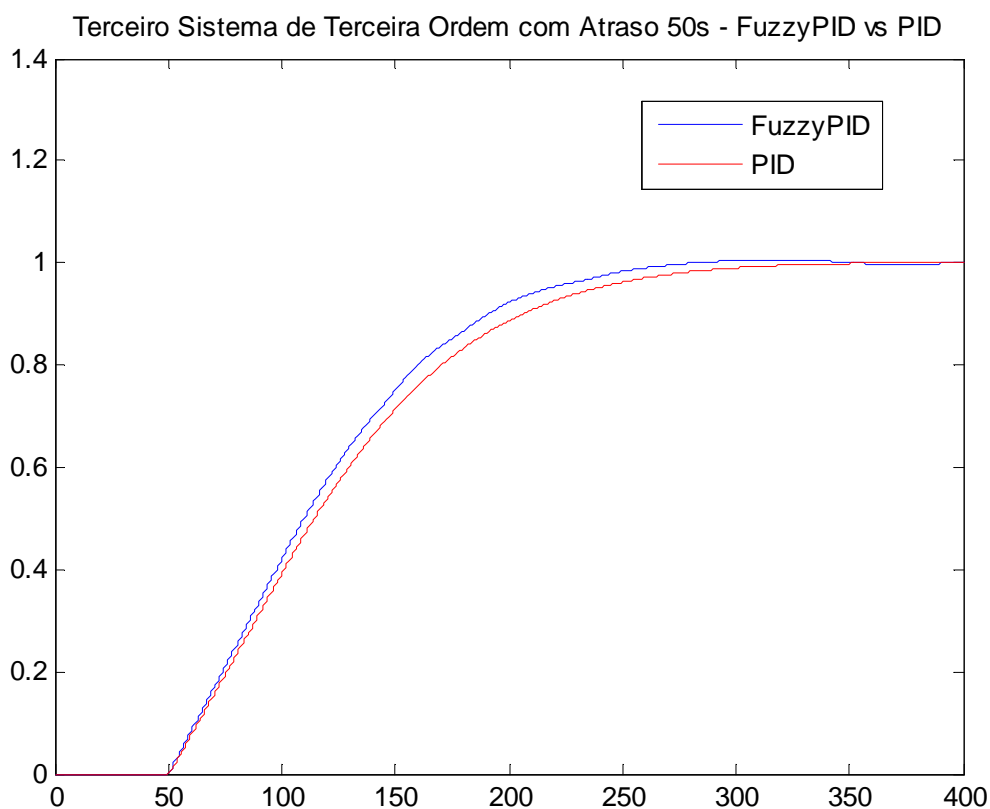


Figura 79 - Análise temporal do terceiro sistema de 3ª ordem com atraso de 50s

Para este terceiro sistema de 3ª ordem com um atraso de 50 segundos a resposta de ambos sistemas de controlo é muito idêntica. Contudo, o controlador FuzzyPID fornece um tempo de subida inferior e o seu tempo de estabelecimento situa-se nos 250 segundos ao contrário do PID que se fixa nos 300 segundos.

Tabela 35 - Valores dos parâmetros temporais do terceiro sistema de 3ª ordem com atraso de 50s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,6 0,6]	[-0,6 0,6]	[-0,6 0,6]	250s
PID	0,1	1,9	0,01	300s

6.2.15. QUARTO SISTEMA DE TERCEIRA ORDEM COM ATRASO 50s

Finalmente, um quarto exemplo de sistemas de 3ª ordem com atraso, neste caso de 50 segundos, descrito pela função de transferência que representa de seguida:

$$TF = \frac{1}{s^3 + 7s^2 + 11s + 5} e^{-50s}$$

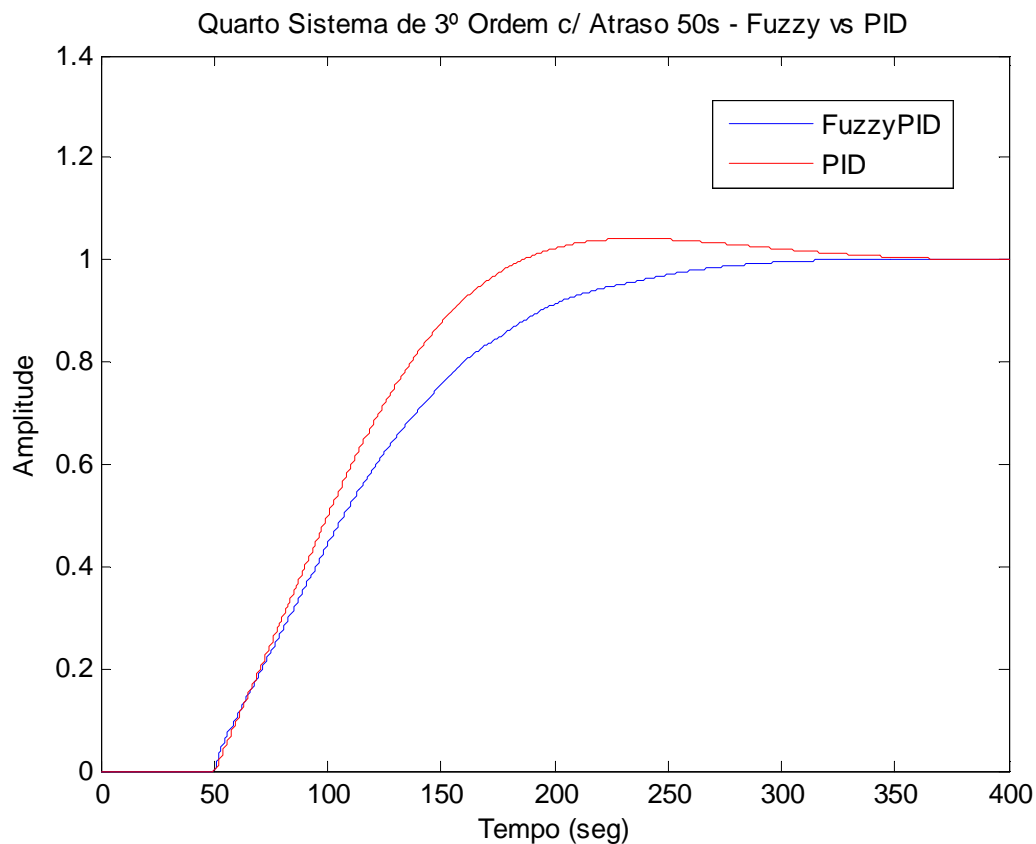


Figura 80 - Análise temporal do quarto sistema de 3ª ordem com atraso de 50s

Neste quarto caso de estudo de sistemas de 3ª ordem, o controlador PID apresenta um tempo de subida inferior, mas por outro lado apresenta *overshoot* e um tempo de estabelecimento superior ao controlador FuzzyPID.

Tabela 36 . Valores dos parâmetros temporais do quarto sistema de 3ª ordem com atraso de 50s

	Kp	Ki	Kd	Tempo de Estabelecimento
FuzzyPID	[-0,6 0,6]	[-0,125 0,125]	[-0,6 0,6]	350s
PID	0,1	0,5	0,01	350s

6.3. CONCLUSÕES

No decorrer das experiências atrás apresentadas verificamos que com o controlador FuzzyPID obtivemos sempre melhores resultados quando comparados com os obtidos com um PID clássico. Mais, tendo por base os diversos tipos de sistemas estudados, comprovamos que os controladores Fuzzy, permitem controlar qualquer tipo de sistema, mesmo quando estes são descritos por funções de transferência complexas, nomeadamente funções de transferência de ordem elevada ou com atrasos na sua resposta. De facto, na indústria o controlo de sistemas com atraso é extremamente difícil de ser conseguido, pois as regras clássicas de sintonia dos controladores não contemplam de forma simples esta situação. No entanto, verificou-se que com o controlador implementado, qualquer uma destas situações foi solucionada com sucesso.

7. CONCLUSÕES

Com o finalizar deste trabalho concluímos que o conceito de Lógica Difusa aplicada ao controlo de sistemas permite que sistemas mais complexos sejam “simplificados” através do conhecimento heurístico que o próprio programador tem sobre o sistema. A possibilidade de se poder definir valores linguísticos ambíguos para se caracterizar situações, como “muito calor” ou “muito alto”, permite um controlo mais cuidado e *userfriendly* do que os controladores usuais.

O controlador PID desempenhou um papel muito importante no desenvolvimento dos controladores, pois é um controlador vulgarmente conhecido por apresentar bons resultados no controlo de sistemas. Este facto fez com que se elevasse a fasquia em termos de performances de controladores.

É importante referir que o *software* de simulação utilizado, o *simulink* do Matlab, o qual representou uma poderosa ferramenta de desenvolvimento de trabalho, sem o qual seria necessário um trabalho árduo para atingir estes resultados. Utilizando o *simulink* foi possível simular o próprio controlador, os sistemas, alterar parâmetros e visualizar as respostas.

Para o controlador FuzzyPID foram criados dois grupos de regras de sintonia para a sintonia dos sistemas com e sem atraso, sendo cada conjunto, composto por três tabelas de

regras para cada componente do PID, K_{P1} , K_{I1} e K_{D1} . Para o controlador PID o método de sintonia utilizado foi o método de sensibilidade limite de Ziegler-Nichols para os sistemas sem atraso e, para os sistemas com atraso foi utilizado o método tentativa erro, partindo dos valores obtidos para o sistema sem atraso e afinando até se obter a melhor resposta possível. A diferença entre os resultados obtidos para os controladores está patente nos gráficos apresentados, sendo que as regras criadas para o controlador FuzzyPID, tanto para os sistemas sem atraso como para os com atraso, superam sem dificuldade o método clássico de sintonia.

Relativamente aos resultados obtidos, visto que estão de acordo com os objectivos e ideias traçadas inicialmente, pode-se concluir de forma global que, tanto para os sistemas sem atraso como para os sistemas com atraso consegue-se melhores resultados com o controlador FuzzyPID do que para o PID. Na sua generalidade os resultados obtidos para o FuzzyPID apresentam um tempo de subida e um tempo de estabelecimento inferior, não apresentam *overshoot* e também muito importante, devido à componente integral o erro em regime permanente é anulado em todos os casos.

Referências Documentais

[BOILER] – Yue Wei-jie, Liu Yong-xin. Boiler Drum Level controlled by Fuzzy Self-adapting PID, College of Electronic Information Engineering, Inner Mongolia University, Hohhot, China.

[JAMES VERNON] – James Vernon. Fuzzy Logic Systems. www.control-systems-principals.co.uk.

[MIRANDA] – Pedro Miranda, Mauro Junior, Diego Kronbauer. Sistema de Controle Difuso de Mamdani Aplicações: Pêndulo Invertido e outras, Departamento de Computação e Estatística, Centro de Ciências Exactas e Tecnologia, Universidade Federal de Mato Grosso do Sul.

[REZNIK] – Leonid Reznik. Fuzzy Controllers, Victoria University of Technology, Melbourne, Australia.

[KEVIN] – Kevin M. Passino, Stephen Yurkovich. Fuzzy Control, Department of Electrical Engineering, The Ohio State University.

[HELLMANN] – M. Hellmann. Fuzzy Logic Introduction. Laboratoire Antennes Radar Telecom, F.R.E CNRS 2272, Equipe Radar Polarimetrie, Université de Rennes 1, UFR S.P.M, Campus de Beaulieu - Bat. 22, 263 Avenue General Leclerc, CS 74205, 35042 Rennes Cedex, France.

[LOURENÇO] – João Lourenço. Sintonia de Controladores PID. Escola Superior de Tecnologia.

[PEREIRA] – Joaquim Jesus Pereira. Sintonia do controlador PID, com Algoritmo de Optimização por Grupo de Partículas. Engenharia Electrotécnica e de Computadores, Universidade de Trás-os-Montes e Alto Douro.

[ARAKI M.] – Araki M. PID Control. Kyoto University, Japan.

[YASSER] – Yasser Mahumud Abdallah. Sintonia de Controlador PID via procedimento adaptativo para controle de atitude veículos lançadores. Engenharia e Tecnologia Espaciais/Mecânica Espacial e Controle, Instituto Nacional de Pesquisas Espaciais, São José dos Campos.

[SANTOS] – Carlos Santos. Análise da Resposta Transitória. Programa de Pós-Graduações em Engenharias de Sistemas Dinâmicos e Energéticos, Universidade Estadual do Oeste do Paraná.

[TROFINO] – Alexandre Trofino. Sistemas Lineares. Departamento de Automação e Sistemas, Centro Tecnológico, Universidade Federal de Santa Catarina.

[ANALISYS] – Understanding Poles and Zeros. Department of Mechanical Engineering, Massachusetts Institute of Technology.

[EI] – Resposta ao degrau de sistema de 2ª ordem. Corpo Docente de EI.

[HELIO] – Prof. Hélio Leães Hey. Análise da Resposta Transitória, do Erro de Regime Permanente e da Estabilidade de Sistemas. Apostila de Sistema de Controle I, Projeto Reenge – Eng. Elétrica.

[MARUYAMA] – Newton Maruyama. O Papel dos Pólos e Zeros, Departamento de Engenharia Mecatrônica, EPUSP, 2007.